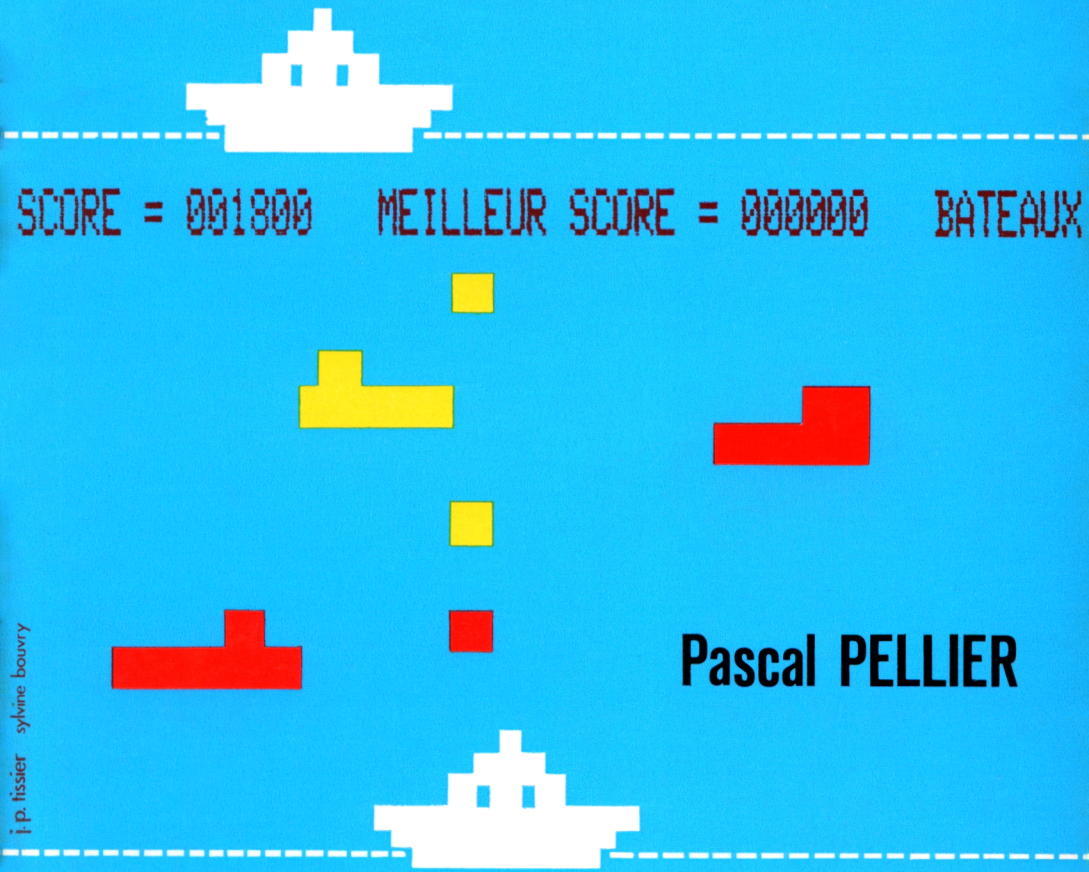
MICRO-ORDINATEURS

PROGRAMMEZ VOS JEUX D'ACTION RAPIDE SUR TRS 80



Pascal PELLIER

j.p. fissier sylvine bouvry

EYROLLES

PROGRAMMEZ VOS JEUX D'ACTION RAPIDE SUR TRS 80

DANS LA MÊME COLLECTION

SCHOMBERG	- Le Basic Universel.
SCHOMBERG	- Micro-ordinateurs : Comment ça marche ?
HERNANDEZ	- Pascal par l'exemple.
ASTIER	- La conduite de l'APPLE II.
NOLLET	- La conduite du ZX 81.
PELLIER	- La conduite du TRS 80.
KOSTAKIS & BAÏKOUR	- Tout sur les disques du TRS 80 (à paraître).
LADEVIE	- Votre gestion avec BASIC sur micro-ordinateur.
QUEINNEC	- Langage d'un autre type : LISP
MONTEIL	- L'assembleur facile du 6502 (à paraître)
LEPAPE	- L'assembleur facile du Z 80 (à paraître)
PELLIER	- Programmez vos jeux d'action rapide sur TRS 80.
OROS & PERBOST	- ZX 81 à la conquête des jeux.

«La loi du 11 mars 1957 n'autorisant, aux termes des alinéas 2 et 3 de l'article 41, d'une part, que les «copies ou reproductions strictement réservés à l'usage privé du copiste et non destinées à une utilisation collective» et, d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, «toute représentation ou reproduction intégrale, ou partielle, faite sans le consentement de l'auteur ou de ses ayants droit ou ayants cause, est illicite» (alinéa 1er de l'article 40)».

«Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait une contrefaçon sanctionnée par les articles 425 et suivants du Code pénal».

© Éditions EYROLLES, 1982

PROGRAMMEZ VOS JEUX D'ACTION RAPIDE SUR TRS 80

par

Pascal PELLIER

Collection animée
par Richard SCHOMBERG

EYROLLES

61, boulevard Saint-Germain - 75005 Paris

1982

Si vous désirez être tenu au courant de nos publications, il vous suffit d'adresser votre carte de visite au :

Service «Presse», Éditions EYROLLES
61, Boulevard Saint-Germain,
75240 PARIS CEDEX 05,

en précisant les domaines qui vous intéressent. Vous recevrez régulièrement un avis de parution des nouveautés en vente chez votre libraire habituel.

Avertissement

Cet ouvrage est destiné à vous faire découvrir les techniques de programmation des jeux d'action rapide afin que vous soyez à même de les réaliser. Il est structuré autour d'un exemple de jeu simulant un combat sous marin. Au fur et à mesure des chapitres vous bâtirez ce programme en commençant par sa structure interne, en réalisant ses graphismes, en le sonorisant, en lui incluant une présentation et enfin en rendant progressive la difficulté. Bien que le jeu soit écrit pour un TRS 80 modèle I ou III avec 4 koctets de mémoire, ce livre sera un bon point de départ pour tous les programmeurs désirant construire des jeux de ce type sur leur propre machine.

Le langage de programmation privilégié pour réaliser ces jeux étant l'assembleur, ce livre s'adresse aux personnes connaissant déjà les principes de base de ce langage. Nous vous conseillons à ce sujet deux ouvrages publiés dans la même collection: le premier intitulé «l'assembleur facile du Z 80» décrit toutes les instructions et les modes de fonctionnement du Z 80, le second qui se nomme «la conduite du TRS 80» montre comment accéder au langage machine sur le TRS 80.

Table des matières

AVERTISSEMENT VII

INTRODUCTION XI

1. Structure générale du programme de jeu 1

1.1. Principes de base 1
1.2. Application à notre exemple 6

2. Sous programmes d'intérêt général 8

2. 1. Multiplication entière : MUL 8
2. 2. Division entière : DIV 10
2. 3. Générateur de nombre aléatoire : RND 11
2. 4. Effacement de l'écran : CLS 13
2. 5. Affichage d'un message sur l'écran : MES 13
2. 6. Conversion d'un nombre binaire entier en une suite de caractères ASCII : TRF 15
2. 7. Lecture d'une ligne de données à partir du clavier : CUR ... 17
2. 8. Sous programmes graphiques : SET, RESET, POINT 19
2. 9. Tracé de droite : DROITE 21
2.10. Sous programme donnant le numéro de ligne correspondant à une case mémoire : LIGNE 24
2.11. Sous programmes divers implantés en ROM 25
2.11.1. RST 18 H 25
2.11.2. CALL 2BH 26
2.11.3. CALL 60 H 26

3. Graphismes 27

3.1. Structure de l'écran 27
3.2. Dessin des graphismes sur l'écran 28
3.2.1. Affichage par caractère 28
3.2.2. Affichage exploitant la structure des caractères 31
3.2.3. Affichage par segment 33
3.3. Déplacement des graphismes 34
3.3.1. Déplacement horizontal 34
3.3.2. Déplacement vertical 35
3.3.3. Déplacement dans une direction quelconque 38
3.3.4. Déplacement par effacement et réécriture systématique de l'écran 41
3.4. Choix des variables et des tableaux nécessaires pour caractériser les personnages 42
3.5. Tests de déplacement 43
3.6. Conseils pour aborder un jeu particulier 44
3.7. Application à notre exemple 44
3.7.1. Déplacement du bateau 46
3.7.2. Déplacement des sous-marins 46
3.7.3. Déplacement des bombes 48
3.7.4. Déplacement des mines 49
3.7.5. Autres sous programmes 50

4. Sonorisation 51

4.1. Production du son 51
4.2. Implantation du son dans le programme de jeu 56
4.3. Application à notre exemple 57

5. Présentation et amélioration du jeu 59

5.1. Présentation 59
5.2. Améliorations 61
5.3. Application à notre exemple 62

6. Méthodologie de réalisation et de mise au point du jeu 65

6.1. Matériel nécessaire et matériel souhaitable 65
6.2. Conseils de réalisation 68
6.3. Conseils de mise au point 70

ANNEXE 1 : **Code des caractères graphiques** 73

ANNEXE 2 : **Structure du clavier** 75

ANNEXE 3 : **SONAR** 77

Introduction

Vous connaissez sans doute ce genre de programmes de jeu qui fait appel à votre habileté et à votre rapidité manuelle. La plupart de ces jeux sont des retranscriptions de jeux existants déjà sur des machines spécialement conçues à cet effet et qui font fureur dans les cafés. Vous vous êtes peut-être demandé comment était-il possible de réaliser sur un micro-ordinateur un programme qui puisse simultanément déplacer différents mobiles (envahisseurs, vaisseaux, bombes, missiles), produire du son sur la sortie magnétophone et détecter les touches appuyées sur le clavier. Si vous avez essayé de réaliser un tel logiciel en BASIC, vous avez du vous apercevoir que c'était tout bonnement impossible. Il n'existe en effet pas de moyen en BASIC de former un son controlable qui produise autre chose qu'un bourdonnement. D'autre part, si on essaye de déplacer plus de trois mobiles en même temps sur l'écran, le déplacement sera tellement lent que le jeu perdra tout son intérêt. La différence entre ce qui peut être réalisé par un programme en BASIC et un véritable jeu commercialisé est tellement flagrante qu'il est difficile de concevoir que ce soit un simple programme qui assure le fonctionnement de ce jeu. C'est pourtant bien un programme qui est à l'origine du jeu, mais qui est écrit dans le langage du microprocesseur (langage machine), seul langage permettant de construire aisément un jeu aussi sophistiqué et aussi rapide en conservant une impression de simultanéité. La vitesse d'exécution étant environ multipliée par un facteur cent, on a largement le temps d'effectuer tous les déplacements désirés et on est même obligé

d'ajouter des boucles d'attente pour ralentir le jeu afin que l'œil puisse discerner les mouvements des personnages sur l'écran. Nous décrirons par conséquent dans ce livre la technique de programmation des jeux en utilisant exclusivement le langage assembleur, en l'occurence celui du Z 80.

Les chapitres qui suivent expliquent les grands principes utilisés pour construire ces jeux et donnent la façon de les appliquer dans un programme de jeu en assembleur que vous allez réaliser au fil des chapitres. Ce jeu que nous avons intitulé SONAR représente un combat entre un navire de guerre piloté par le joueur et une multitude de sous-marins qui défilent sous lui en larguant des mines. Le navire doit se déplacer pour éviter celles-ci et doit abattre le maximum de sous-marins pour gagner des points et établir ainsi le meilleur score. Il peut se déplacer à droite ou à gauche de l'écran grâce aux touches → et ← et lacher les bombes par la proue au la poupe à l'aide des touches Q et W. Les sous-marins se déplacent horizontalement d'un côté à l'autre de l'écran à une vitesse constante qui peut cependant varier d'un sous-marin à un autre. Les plus rapides rapportent 400 points quand ils sont abattus tandis que les plus lents n'en donnent que 100. Quand un sous-marin est détruit par une des bombes lachées par le bateau du joueur le nombre de points gagnés est inscrit un court instant à l'endroit où le sous-marin a explosé. Les mines lachées par les sous-marins sont représentées par des étoiles et se déplacent vers le haut jusqu'à la surface où elles explosent en produisant une gerbe d'eau. Si elles rencontrent le vaisseau du joueur, celui-ci sombre complétement. D'autre part ces mines sont supposées se déplacer dans un plan situé en arrière, aussi elles peuvent être masquées par un objet (sous-marin, bombe) passant devant, ce qui accroit la difficulté du jeu. Le déroulement du jeu peut être interrompu à tout moment par l'appui sur CLEAR et BREAK. Au cours d'une partie, le joueur dispose de trois navires, plus un navire supplémentaire s'il atteint 20 000 points. Son score, ainsi que le meilleur score et le nombre de bateaux restants sont affichés en permanence en haut de l'écran. Le jeu est entièrement sonorisé et est doté d'une présentation succinte incluant un mode démonstration ayant pour but de montrer le déroulement d'une partie. Il occupe environ 2,5 koctets de mémoire et peut s'exécuter sur un TRS 80 modèle I ou III avec 4 koctets seulement.

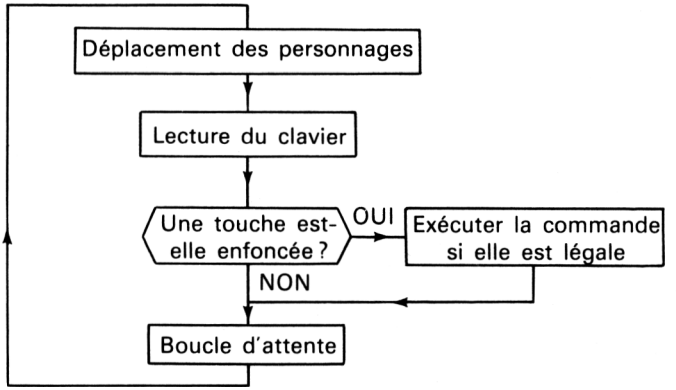
1

Structure générale du programme de jeu

1.1. PRINCIPES DE BASE

Vous avez sans doute déjà programmé en BASIC et vous vous êtes rendu compte des problèmes de lenteur inhérents à ce langage. Peut-être avez-vous essayé de faire des jeux d'animation en BASIC et vous avez du constaté qu'il était difficile de faire se mouvoir plusieurs personnages en même temps tout en conservant un jeu rapide. Pour arriver à cette prouesse il vous a fallu minimiser au maximum les instructions afin de diminuer la lenteur d'exécution. En passant au langage assembleur, ces problèmes se trouvent immédiatement annulés. Aussi, l'esprit de programmation est totalement différent. Il vaut mieux faire exécuter des instructions inutiles plutôt que de se casser la tête à chercher les tests à faire pour les éviter. Il est préférable de faire travailler le TRS 80 qui a du temps à perdre plutôt que le programmeur. Aussi ne vous occupez plus du gain de temps que vous imposait le BASIC et laissez libre court à votre imagination. Citons un exemple de ce principe utilisé dans un programme commercialisé. L'explosion du vaisseau dirigé par le joueur s'effectue en allumant des segments graphiques tout autour de celui-ci pour simuler l'éclatement de l'appareil. A la fin de l'explosion, il faudra effacer les éclats et, plutôt que de trouver un algorithme qui le fasse, il est plus simple de stocker tout l'écran dans une mémoire annexe de 1 koctet avant l'explosion et de restituer l'image après l'explosion à partir de cette mémoire.

Les jeux d'animation décrits dans ce livre sont dits en "temps réel", en ce sens que le programme se déroule de façon naturelle même si aucun joueur n'exerce une action sur le clavier. Ce terme dérivé des machines plus importantes désigne un programme qui s'exécute en monopolisant totalement l'ordinateur contrairement aux programmes ordinaires qui fonctionnent en "temps partagé" c'est-à-dire que, pendant les inévitables temps morts d'un programme dus à des lectures sur le clavier ou sur un périphérique, on permet le déroulement d'autres programmes de façon à augmenter le pourcentage d'activité du processeur. Ce terme perd son sens sur un micro-ordinateur où tous les programmes s'exécutent en temps réel, mais par analogie il désigne un programme qui ne reste pas bloqué pendant un temps indéterminé en attendant une réponse d'un agent extérieur. Il est donc exclus dans notre application d'utiliser des fonctions comme INPUT du BASIC, qui ont pour effet d'attendre que l'utilisateur veuille bien appuyer sur ENTER pour saisir ses commandes. Pour donner l'impression que le programme se déroule quasiment indépendemment du joueur, celui-ci ne pouvant modifier que certains paramètres, il faudra au contraire effectuer une lecture du clavier à intervalles de temps régulièrement espacés. Les intervalles de temps devront être suffisemment brefs pour donner l'illusion que le programme effectue des lectures du clavier de façon permanente. Ces opérations de lecture consistent à déterminer la ou les touches enfoncées en examinant les valeurs contenues dans les mémoires réalisant l'interface entre le TRS 80 et le clavier. Si une touche est enfoncée, on pourra alors exécuter la commande du joueur à condition qu'elle soit légale. Entre deux lectures du clavier on aura tout le temps de faire les calculs et les traitements nécessaires pour déplacer les personnages sur l'écran et produire les sonorités adéquates. Le microprocesseur étant tellement rapide, il faudra même inclure une boucle d'attente afin que le jeu ne soit pas trop difficile. Pour pouvoir satisfaire aux caractéristiques citées précédemment, le programme aura la structure d'une boucle sans fin. A chaque passage dans la boucle on fera successivement le déplacement des personnages, la lecture du clavier avec l'exécution des commandes qui en résulte, et enfin une boucle d'attente. Ceci peut être résumé par l'organigramme élémentaire suivant :

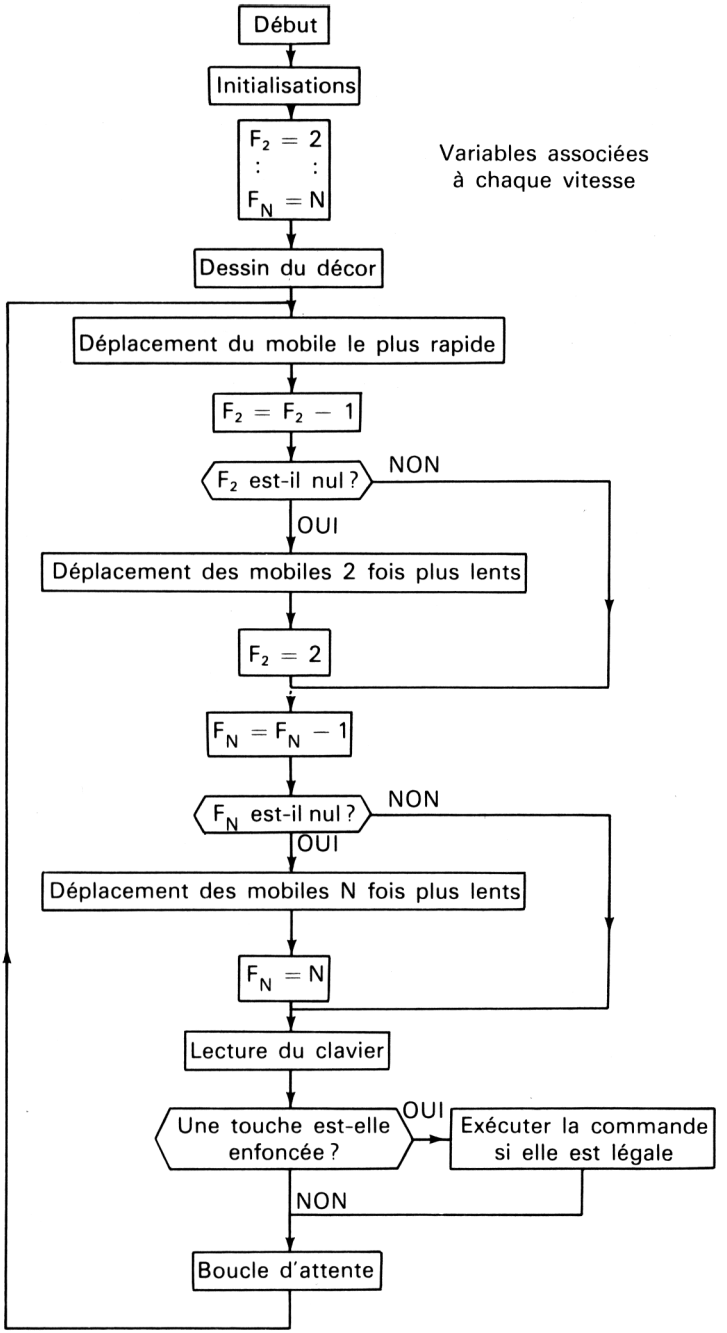


Le compteur utilisé dans la boucle d'attente devra être initialisé avec une valeur judicieusement choisie. Cette valeur ne devra pas être trop petite pour que la difficulté du jeu soit raisonnable et pas trop grande pour donner l'impression que les différentes actions s'effectuent simultanément et de manière continue.

Pour déplacer un personnage, il suffit à chaque passage dans la boucle principale de l'effacer de sa position initiale et de le redessiner dans une position voisine. Si l'intervalle entre deux positions consécutives est suffisemment faible, l'œil du joueur ne s'apercevra pas que le déplacement se fait par à coup et croira qu'il est continu. Les déplacements des divers mobiles ne se font pas tous nécessairement à la même vitesse. Pour assurer cet effet, on prendra comme base de temps pour la boucle principale l'intervalle entre deux déplacements du plus rapide des personnages. Pour simplifier le problème nous supposerons que l'intervalle entre deux déplacements des autres mobiles sera un multiple de cette base de temps. Ainsi le mobile le plus rapide sera bougé à chaque passage dans la boucle, un mobile deux fois plus lent sera déplacé une fois sur deux, un mobile N fois plus lent passera dans la boucle N – 1 fois avant d'être déplacé. Une variable sera affectée à chaque type de vitesse pour savoir à quel moment doit être effectué le déplacement. Cette variable initialisée par le rapport de vitesse N, sera décrémentée à chaque passage dans la boucle. Si elle est nulle le déplacement sera effectué et la variable sera réinitialisée à la valeur N. Il est possible de raffiner

cette procédure en prenant une base de temps inférieure à l'intervalle séparant deux positions voisines du plus rapide des mobiles. On peut ainsi obtenir en principe n'importe quel rapport de vitesse entre les différents personnages. Toutefois la première solution est souvent suffisante, la seconde peut s'employer à la rigueur pour les fractions valant $\frac{1}{2}$ ou $\frac{1}{3}$ mais il est bien rare de descendre au delà.

Dans le programme de jeu final on commencera l'exécution en effectuant plusieurs initialisations et en dessinant le décor du jeu sur l'écran. Ensuite on se branchera au début de la boucle principale. La sortie de la boucle se produira lorsque la partie sera terminée en testant la valeur des variables qui limitent le jeu (nombre de vaisseaux, temps, etc...). L'organigramme général du jeu s'apparentera donc au suivant :


Début
Initialisations
F2 = 2
:
FN = N
Variables associées à chaque vitesse
Dessin du décor
Déplacement du mobile le plus rapide
F2 = F2 − 1
F2 est-il nul ?
NON
OUI
Déplacement des mobiles 2 fois plus lents
F2 = 2
FN = FN − 1
FN est-il nul ?
NON
OUI
Déplacement des mobiles N fois plus lents
FN = N
Lecture du clavier
Une touche est-elle enfoncée ?
OUI
Exécuter la commande si elle est légale
NON
Boucle d'attente

1.2. APPLICATION A NOTRE EXEMPLE

Avant de commencer à programmer quoi que ce soit, il faut analyser soigneusement le problème à traiter de façon à savoir quels sont les déplacements et les tests nécessaires et ceux qui pourrait être évités. Ensuite il faut choisir à priori les rapports de vitesse entre les différents mobiles. Ces rapports pourront être ajustés par la suite par expérimentation directe sur le programme en fonctionnement. Dans notre exemple nous avons dénombré quatre types de mobiles à savoir: le bateau du joueur, les sous-marins, les bombes anti-sous-marines et les mines larguées par les sous-marins. L'unité de temps choisie est égale à la durée d'un déplacement élémentaire du bateau. La vitesse des sous-marins peut être de un à quatre fois plus petite que celle du bateau, aussi nous utiliserons un sous programme général appelé à chaque passage dans la boucle principale pour effectuer le déplacement des sous-marins et gérer les rapports de vitesse entre eux. Les bombes constituées de deux segments graphiques côte à côte sont deux fois plus lentes que le bateau. Par conséquent elles sont déplacées une fois sur deux grâce à la variable F1 initialisée à deux. Les mines vont à la même vitesse mais, étant donné qu'elles se déplacent d'un caractère à chaque fois au lieu d'un segment pour les bombes, elles sont bougées trois fois moins souvent à l'aide de la variable FØ initialisée à 3.

Pour conserver le schéma de la structure générale du jeu, il est préférable d'organiser chaque type d'action en sous programme. Le programme principal sera constitué essentiellement d'appels à des sous programmes exécutant chacun une action élémentaire bien définie, ce qui permettra de disposer en une seule page de listing de toute l'ossature du jeu. Dans notre exemple, nous avons d'autre part délibérément distingué l'action faisant naître un mobile (tir, sortie aléatoire) du déplacement proprement dit de celui-ci, de sorte que chaque sous programme ne soit pas trop gros. Le listing qui suit représente la boucle principale du jeu SONAR. Chaque sous programme est référencé par une étiquette de trois caractères qui sont les initiales de l'action produite.

```
00010 Z0      CALL    ASM       ;AVANCE DES SOUS-MARINS
00020         CALL    ABA       ;AVANCE DU BATEAU
00030         LD      A,(F1)    ;PREMIERE VARIABLE DE DEPLACEMENT
00040         DEC     A         ;DECREMENTER CETTE VARIABLE
00050         JR      NZ,Z2     ;SI NON NULLE NE RIEN FAIRE
00060 ;
00070 ;CETTE PARTIE EST EXECUTEE UNE FOIS SUR DEUX
00080 ;
00090         CALL    ABM       ;AVANCE DES BOMBES
00100         LD      A,(F0)    ;SECONDE VARIABLE DE DEPLACEMENT
00110         DEC     A         ;DECREMENTER CETTE VARIABLE
00120         JR      NZ,Z1     ;SI NON NULLE NE RIEN FAIRE
00130 ;
00140 ;CETTE PARTIE EST EXECUTEE UNE FOIS SUR 6
00150 ;
00160         CALL    TSM       ;TIR DES SOUS-MARINS
00170         CALL    AMN       ;AVANCE DES MINES
00180         CALL    TBA       ;TIR DU BATEAU
00190         CALL    SSM       ;SORTIE DES SOUS-MARINS
00200         LD      A,3       ;REINITIALISER VARIABLE F0
00210 Z1      LD      (F0),A    ;NOUVELLE VALEUR
00220         LD      A,2       ;REINITIALISER VARIABLE F1
00230 Z2      LD      (F1),A    ;NOUVELLE VALEUR
00240         LD      BC,1000H  ;COMPTEUR DE BOUCLE D'ATTENTE
00250         CALL    60H       ;ATTENTE
00260         JR      Z0        ;BOUCLER
```

Les trois sous programmes utilisés pour générer les mobiles (TSM, TBA, SSM) ont été placé dans la partie de la boucle la moins souvent atteinte afin de limiter le nombre de personnages sur l'écran. Il aurait été tout à fait possible de les placer juste à côté du sous programme déplaçant ces mobiles.

2

Sous-programmes d'intérêt général

Nous allons décrire dans ce chapitre une série de sous programmes utilitaires qui faciliteront la construction des programmes de jeu. Ces sous programmes permettent l'utilisation de certaines fonctions fournies par le BASIC et qui sont malheureusement inexistantes en assembleur. Leur emploi ne se limite pas au domaine des jeux, ils seront certainement appréciés dans des applications plus sérieuses. La plupart de ces sous programmes existent dans la ROM du BASIC. Toutefois ils sont souvent assez lents et leur implantation diffère selon que l'on utilise un modèle I ou un modèle III. Aussi il est préférable pour des questions de compatibilité et de rapidité de créer ses propres sous programmes, d'autant plus qu'ils ne devraient pas occuper beaucoup de place dans un logiciel de jeu.

Pour chaque utilitaire organisé en sous programme nous donnerons une explication sur son fonctionnement, son mode d'emploi ainsi que le listing d'assemblage. Celui-ci sera toujours implanté en zéro pour des raisons d'homogénéité. Il sera par ailleurs abondemment commenté.

2.1. MULTIPLICATION ENTIÈRE : MUL

Ce programme emploie un procédé analogue à celui que nous utilisons lorsque nous effectuons une multiplication en binaire à la main.

Prenons comme exemple la multiplication de 1010 par 1100 on obtient une opération de ce type :

1010	Multiplicande
× 1100	Multiplicateur
0000	
0000	produits partiels
1010	
1010	
1111000	Résultat

Les quatre produits partiels dans notre exemple valent soit 0000 soit 1010 suivant que le bit correspondant du multiplicateur vaut 0 ou 1. Le résultat sera obtenu en faisant la somme des divers produits partiels convenablement décalés. Nous nous sommes ramenés à des opérations de tests, d'addition et de décalage que le microprocesseur sait effectuer. L'algorithme utilisé dans le programme découle de ce que nous venons de dire, en procédant toutefois en sens inverse pour des raisons de commodité. On commence par traiter les bits de poids fort du multiplicateur et on décale à chaque fois le résultat. Si le bit du multiplicateur est à 1 on ajoute le multiplicande, sinon on ne fait rien. L'opération est terminée quand le bit de plus faible poids a été traité.

```
                00010 ;*******************************************************
                00020 ;
                00030 ; SOUS PROGRAMME DE MULTIPLICATION ENTIERE
                00040 ; D'UN NOMBRE DE 16 BITS PAR UN NOMBRE DE 8 BITS
                00050 ;
                00060 ;  EN ENTREE: DE CONTIENT LE MULTIPLICANDE
                00070 ;              A  CONTIENT LE MULTIPLICATEUR
                00080 ;
                00090 ;  EN SORTIE: HL CONTIENT LE RESULTAT DE LA MULTIPLICATION
                00100 ;
                00110 ;*******************************************************
                00120 ;
0000 210000     00130 MUL     LD      HL,0     ;INITIALISATION DU RESULTAT
0003 29         00140 M0      ADD     HL,HL    ;LE RESULTAT EST MULTIPLIE PAR 2
0004 CB27       00150         SLA     A        ;LE BIT DE POIDS FORT DE A EST
                00160                          ;MIS DANS LE CARRY
0006 3001       00170         JR      NC,M1    ;SI CE BIT VAUT 0
0008 19         00180         ADD     HL,DE    ;SINON AJOUTER LE MULTIPLICANDE
0009 20F8       00190 M1      JR      NZ,M0    ;SI A N'EST PAS NUL PASSER AU BIT
                00200                          ;SUIVANT
000B C9         00210         RET              ;SINON FIN DU SOUS PROGRAMME
0000            00220         END
00000 TOTAL ERRORS
```

Le programme effectue une multiplication d'un nombre de 16 bits par un nombre de 8 bits et donne un résultat sur 16 bits. Aussi si les nombres sont trop grands, il se peut que le résultat ne tienne plus sur 16 bits, il y a alors débordement et le résultat est erroné. Par conséquent, il faudra s'assurer que ce cas ne pourra pas se produire quand vous emploierez cet utilitaire de multiplication.

2.2. DIVISION ENTIÈRE: DIV

L'algorithme utilisé dans ce sous programme découle du procédé employé quand on effectue une division binaire à la main. Prenons comme exemple la division de 1011011 par 1110:

dividende	1011011	1110	diviseur
restes partiels	10110	0110	quotient
	10001		
reste	00111		

Le quotient vaut 110 et le reste 111.

Le procédé utilisé dans le programme est le suivant: Le registre A destiné à recevoir les restes partiels est initialisé à zéro.

On met le bit de poids fort du dividende dans le bit 0 de A. On regarde alors si A est supérieur au dividende. S'il ne l'est pas on place un bit 0 dans le quotient, sinon on place un bit 1 et on soustrait le diviseur de A. Ensuite A est décalé vers la gauche et on recommence la procédure jusqu'à ce que tous les bits du dividende soient traités.

Exemple:

	valeur de A	quotient	
cycle 0	0		
cycle 1	1	0	A est inférieur au diviseur
cycle 2	10	0	
cycle 3	101	0	
cycle 4	1011	0	
cycle 5	10110	1	A est supérieur au diviseur
cycle 6	10001	1	A est supérieur au diviseur
cycle 7	111	0	A est inférieur au diviseur

La valeur du quotient lue de haut en bas est égale à 0000110 et le reste est contenu dans A (111)

```
                00010 ;*****************************************************
                00020 ;
                00030 ; SOUS PROGRAMME DE DIVISION ENTIERE
                00040 ; D'UN NOMBRE DE 16 BITS PAR UN NOMBRE DE 8 BITS
                00050 ;
                00060 ;  EN ENTREE: HL CONTIENT LE DIVIDENDE
                00070 ;             C  CONTIENT LE DIVISEUR
                00080 ;
                00090 ;  EN SORTIE: HL CONTIENT LE QUOTIENT
                00100 ;             A  CONTIENT LE RESTE
                00110 ;
                00120 ;*****************************************************
                00130 ;
0000 AF         00140 DIV     XOR     A       ;A VA CONTENIR LES PREMIERS BITS
                00150                         ;DU DIVIDENDE
0001 0610       00160         LD      B,16    ;NOMBRE DE BITS DU DIVIDENDE
0003 29         00170 H2      ADD     HL,HL   ;LE BIT 16 DU DIVIDENDE EST MIS
                00180                         ;DANS LE CARRY
0004 17         00190         RLA             ;CE BIT EST MIS EN POIDS FAIBLE
                00200                         ;DANS A
0005 3803       00210         JR      C,H4    ;SI DEBORDEMENT DE A, A EST
                00220                         ;SUPERIEUR AU DIVISEUR
0007 B9         00230         CP      C       ;A EST-IL SUPERIEUR AU DIVISEUR?
0008 3802       00240         JR      C,H3    ;NON PASSER AU BIT SUIVANT
000A 91         00250 H4      SUB     C       ;SOUSTRAIRE LE DIVISEUR
000B 2C         00260         INC     L       ;1 DANS LE QUOTIENT
000C 10F5       00270 H3      DJNZ    H2      ;BOUCLE POUR TRAITER TOUT LES BITS
000E C9         00280         RET             ;FIN DU SOUS PROGRAMME
0000            00290         END
00000 TOTAL ERRORS
```

Le programme effectue une division d'un nombre de 16 bits par un nombre de 8 bits, range le quotient dans un nombre de 16 bits et le reste dans un nombre de 8 bits. Le seul cas où peut se produire un débordement correspond au cas où le diviseur (C) est nul.

2.3. GÉNÉRATEUR DE NOMBRE ALÉATOIRE: RND

Ce sous programme calcule un nombre aléatoire de 8 bits dont la valeur est comprise entre 1 et un nombre maximal fourni lors de l'appel du sous programme. Celui-ci commence par déterminer un nombre aléatoire compris entre 0 et 255, en utilisant le registre REFRESH du Z 80 et la variable SEED modifiée à chaque appel. Le registre R contient un nombre pseudo-aléatoire compris entre 0 et 127, aussi il faut effectuer un petit traitement pour obtenir un nombre variant entre 0 et 255. Une fois ce nombre connu, on le multiplie avec le nombre maximal donné en entrée. En ne gardant que l'octet de poids fort du résultat auquel on ajoute 1, on obtient le nombre aléatoire désiré. La fonction RANDOM doit être appelée au début du programme pour initialiser la variable SEED.

```
                00010 ;*****************************************************
                00020 ;
                00030 ; INITIALISATION DU GENERATEUR ALEATOIRE
                00040 ; FONCTION RANDOM
                00050 ;
                00060 ;*****************************************************
                00070 ;
0000 ED5F       00080 RANDOM  LD      A,R     ;NOMBRE ALEATOIRE
0002 322E00     00090         LD      (SEED),A;INITIALISATION DU SEED
0005 C9         00100         RET
                00110 ;*****************************************************
                00120 ;
                00130 ; GENERATEUR DE NOMBRE ALEATOIRE
                00140 ;
                00150 ; EN ENTREE: A CONTIENT LA LIMITE MAXIMALE DU NOMBRE
                00160 ;                ALEATOIRE
                00170 ;
                00180 ; EN SORTIE: A CONTIENT UN NOMBRE ALEATOIRE COMPRIS ENTRE
                00190 ;                1 ET LE NOMBRE DONNE EN ENTREE
                00200 ;
                00210 ; TOUS LES REGISTRES SONT SAUVEGARDES SAUF AF
                00220 ;
                00230 ;*****************************************************
                00240 ;
0006 D5         00250 RND     PUSH    DE      ;SAUVEGARDE DES REGISTRES DE
0007 E5         00260         PUSH    HL      ;ET HL
0008 1600       00270         LD      D,0     ;LE NOMBRE MAXIMAL EST PLACE
000A 5F         00280         LD      E,A     ;DANS DE
000B ED5F       00290         LD      A,R     ;LECTURE D'UN NOMBRE ALEATOIRE
                00300                         ;CONTENU DANS LE REGISTRE REFRESH
                00310                         ;A EST COMPRIS ENTRE 0 ET 127
000D 6F         00320         LD      L,A     ;CE NOMBRE EST SAUVEGARDE DANS L
000E 3A2E00     00330         LD      A,(SEED);LECTURE DU SEED PRECEDENT
0011 AD         00340         XOR     L       ;CALCUL DU SEED SUIVANT
0012 17         00350         RLA             ;A VARIE ENTRE 0 ET 254
0013 6F         00360         LD      L,A     ;SAUVEGARDE DANS L
0014 ED5F       00370         LD      A,R     ;NOMBRE ALEATOIRE
0016 AD         00380         XOR     L       ;CALCUL DU SEED SUIVANT
0017 322E00     00390         LD      (SEED),A;NOUVEAU SEED
001A CD2200     00400         CALL    MUL     ;MULTIPLIER LE NOMBRE ALEATOIRE
                00410                         ;AVEC LE NOMBRE MAXIMAL DONNE
001D 7C         00420         LD      A,H     ;NOMBRE ALEATOIRE ENTRE 0 ET
                00430                         ;LE NOMBRE MAXIMAL MOINS 1
001E 3C         00440         INC     A       ;NOMBRE ALEATOIRE ENTRE 1 ET
                00450                         ;LE NOMBRE MAXIMAL
001F E1         00460         POP     HL      ;RESTAURATION DES REGISTRES DE
0020 D1         00470         POP     DE      ;ET HL
0021 C9         00480         RET
                00490 ;
                00500 ; SOUS PROGRAMME DE MULTIPLICATION
                00510 ;
0022 210000     00520 MUL     LD      HL,0
0025 29         00530 H0      ADD     HL,HL
0026 CB27       00540         SLA     A
0028 3001       00550         JR      NC,H1
002A 19         00560         ADD     HL,DE
002B 20F8       00570 H1      JR      NZ,H0
002D C9         00580         RET
0001            00590 SEED    DEFS    1
0000            00600         END
00000 TOTAL ERRORS
```

2.4. EFFACEMENT DE L'ÉCRAN : CLS

Pour effacer l'écran, il faut mettre un caractère "espace" dans toutes les cases de la mémoire écran. C'est ce que nous faisons en employant l'instruction LDIR qui est très rapide. Le caractère "espace" peut être obtenu de deux façons : en mode alphanumérique par le code 20 H ou en mode graphique par le code 80 H. Étant donné que nous allons travailler en mode graphique, nous avons préféré utiliser le caractère 80 H.

```
                00010 ;*******************************************************
                00020 ;
                00030 ; EFFACEMENT DE L'ECRAN
                00040 ;
                00050 ;*******************************************************
                00060 ;
0000 21003C     00070 CLS     LD      HL,3C00H ;DEBUT DE LA MEMOIRE ECRAN
0003 11013C     00080         LD      DE,3C01H ;DEBUT DE LA MEMOIRE ECRAN +1
0006 01FF03     00090         LD      BC,1023  ;TAILLE DE LA MEMOIRE ECRAN -1
0009 3680       00100         LD      (HL),80H ;EFFACEMENT DU PREMIER CARACTERE
000B EDB0       00110         LDIR             ;EFFACEMENT DE TOUT L'ECRAN
000D C9         00120         RET
0000            00130         END
00000 TOTAL ERRORS
```

2.5. AFFICHAGE D'UN MESSAGE SUR L'ÉCRAN : MES

Voici un sous programme visant à standardiser l'affichage des messages sur l'écran. Il a pour effet de placer, à partir de l'adresse contenue dans DE, une suite de caractères de longueur quelconque terminée par un zéro et dont l'adresse initiale est dans HL. Il présente la particularité de tenir compte de la présence des minuscules sur le TRS 80. Sur la plupart des modèles I le bit 6 de la mémoire écran est inexistant : aussi quand on écrit une minuscule on voit apparaître un autre caractère spécial. Il est donc nécessaire de convertir les minuscules en majuscules. C'est ce que fait le sous programme MIN. Ce dernier utilise le résultat du sous programme PRMN qui doit être exécuté au début du programme afin de tester la présence ou l'absence des minuscules.

```
00010 ;*******************************************************
00020 ;
00030 ; DETERMINATION DE LA PRESENCE DES MINUSCULES
00040 ;
00050 ; FLAG VAUT 0 S'IL N'Y A PAS DE MINUSCULES
00060 ;      ET 0AFH DANS LE CAS CONTRAIRE
00070 ;
00080 ;*******************************************************
```

```
                00090 ;
0000 21003C     00100 PRMN    LD      HL,3C00H ;HL POINTE VERS LA PREMIERE CASE
                00110                          ;DE L'ECRAN
0003 3661       00120         LD      (HL),61H ;LA MINUSCULE a Y EST PLACEE
0005 7E         00130         LD      A,(HL)   ;LECTURE DE LA VALEUR MEMORISEE
0006 FE61       00140         CP      61H      ;EST-CE BIEN a ?
0008 2001       00150         JR      NZ,OE+1  ;SI NON METTRE FLAG A ZERO
                00160                          ;EN EXECUTANT XOR  A (CODE 0AFH)
000A 3EAF       00170 OE      LD      A,0AFH   ;METTRE LE FLAG A 0AFH
000C 322B00     00180         LD      (FLAG),A ;POSITIONNEMENT DU FLAG
000F C9         00190         RET
                00200 ;*****************************************************
                00210 ;
                00220 ; AFFICHAGE D'UN MESSAGE SUR L'ECRAN EN TENANT COMPTE
                00230 ; DE LA PRESENCE OU DE L'ABSENCE DES MINUSCULES
                00240 ;
                00250 ; EN ENTREE: HL CONTIENT L'ADRESSE DU PREMIER ELEMENT
                00260 ;               DU MESSAGE A AFFICHER
                00270 ;               LE MESSAGE DOIT SE TERMINER PAR UN OCTET 0
                00280 ;            DE CONTIENT L'ADRESSE A PARTIR DE LAQUELLE
                00290 ;               IL DOIT ETRE POSITIONNE
                00300 ;
                00310 ; LES REGISTRES HL, DE ET AF SONT MODIFIES
                00320 ;
                00330 ;*****************************************************
                00340 ;
0010 7E         00350 MES     LD      A,(HL)  ;LECTURE DU CARACTERE A AFFICHER
0011 B7         00360         OR      A       ;EST-CE UN ZERO ?
0012 C8         00370         RET     Z       ;SI OUI LE MESSAGE EST TERMINE
0013 CD1B00     00380         CALL    MIN     ;CONVERSION EN MAJUSCULE SI LES
                00390                         ;MINUSCULES NE SONT PAS INSTALLEES
0016 12         00400         LD      (DE),A  ;AFFICHAGE D'UN CARACTERE
0017 13         00410         INC     DE      ;INCREMENTATION DES POINTEURS DE
0018 23         00420         INC     HL      ;ET HL
0019 18F5       00430         JR      MES     ;PASSER AU CARACTERE SUIVANT
                00440 ;*****************************************************
                00450 ;
                00460 ; SOUS PROGRAMME TRANSFORMANT UNE MINUSCULE EN MAJUSCULE
                00470 ; SI LES MINUSCULES NE SONT PAS INSTALLEES SUR LE TRS 80
                00480 ;
                00490 ; EN ENTREE: A CONTIENT LE CARACTERE A TESTER
                00500 ;
                00510 ; EN SORTIE: A CONTIENT LE CARACTERE MODIFIE
                00520 ;
                00530 ;*****************************************************
                00540 ;
001B FE7B       00550 MIN     CP      7BH     ;LE CODE ASCII EST-IL SUPERIEUR
                00560                         ;A CELUI DE z ?
001D D0         00570         RET     NC      ;SI OUI RETOUR
001E FE61       00580         CP      61H     ;LE CODE EST-IL INFERIEUR A a ?
0020 D8         00590         RET     C       ;SI OUI RETOUR
0021 47         00600         LD      B,A     ;SAUVEGARDE DANS B
0022 3A2B00     00610         LD      A,(FLAG);LECTURE DU FLAG
0025 B7         00620         OR      A       ;EST-IL NUL ?
0026 78         00630         LD      A,B     ;RESTAURER LE CARACTERE
0027 C0         00640         RET     NZ      ;SI FLAG<>0 RETOUR (MINUSCULES)
0028 CBAF       00650         RES     5,A     ;CONVERSION EN MAJUSCULE
002A C9         00660         RET
0001            00670 FLAG    DEFS    1
0000            00680         END
00000 TOTAL ERRORS
```

Pour appeler cet utilitaire, on procédera comme suit :

— Tester la présence des minuscules au début de programme en exécutant : CALL PRMN

— Chaque fois que vous aurez un message à afficher, utilisez une séquence de ce type :

```
LD      HL, BONJ
LD      DE, 3COOH
CALL    MES
```

où l'étiquette BONJ sera définie par les directives DEFM et DEFB.

```
BONJ    DEFM    'Bonjour'
        DEFB    Ø
```

2.6. CONVERSION D'UN NOMBRE BINAIRE ENTIER EN UNE SUITE DE CARACTÈRES ASCII : TRF

Ce sous programme sera très utile pour afficher les scores sur l'écran pendant le jeu. Chaque fois que le score sera modifié, on l'appelera pour réécrire le nouveau score. Cet utilitaire emploie une procédure simple, mais qui a l'avantage d'être très rapide. Il détermine successivement tous les chiffres du nombre à traduire en commençant par le chiffre le plus significatif. Ceci se fait en divisant successivement le nombre par toutes les puissances de 10, en commençant par 10 000 et en allant jusqu'à 10. Le quotient transformé en ASCII donne un des chiffres souhaité, le reste est utilisé comme nouvel argument pour effectuer les autres divisions :

```
Nombre   = chiffre 1 * 10000 + reste 1
reste 1  = chiffre 2 * 1000  + reste 2
reste 2  = chiffre 3 * 100   + reste 3
reste 3  = chiffre 4 * 10    + reste 4
reste 4  = chiffre 5
```

Les chiffres ainsi calculés sont placés les uns à la suite des autres dans une zone réservée appelée BUF. Lors de la dernière division on range le nombre des unités, on place un zéro à la fin de la chaine et on

positionne HL au début. On pourra alors utiliser le sous programme MES pour l'affichage sur l'écran.

```
                  00010 ;*****************************************************
                  00020 ;
                  00030 ; TRANSFORMATION D'UN NOMBRE ENTIER SIGNE CODE SUR 16
                  00040 ; BITS EN UNE SUITE DE CARACTERES ASCII LE REPRESENTANT
                  00050 ; LE NOMBRE PEUT VARIER ENTRE -32768 ET 32767
                  00060 ;
                  00070 ; POUR OBTENIR UNE CONVERSION SUR UN ENTIER NON SIGNE
                  00080 ; DONT LA VALEUR EST COMPRISE ENTRE 0 ET 65535 IL SUFFIT
                  00090 ; DE DETRUIRE LES LIGNES 230 A 300
                  00100 ;
                  00110 ; EN ENTREE: HL CONTIENT LE NOMBRE A CONVERTIR
                  00120 ;
                  00130 ; EN SORTIE: HL CONTIENT L'ADRESSE DU PREMIER ELEMENT DE
                  00140 ;              LA CHAINE REPRESENTANT LE NOMBRE
                  00150 ;              IL EST ALORS POSSIBLE D'UTILISER LE SOUS
                  00160 ;              PROGRAMME MES POUR L'AFFICHAGE
                  00170 ;
                  00180 ;*****************************************************
                  00190 ;
0000 DD215000     00200 TRF     LD      IX,BUF+1    ;IX SERT A REMPLIR LE BUFFER
                  00210                             ;QUI VA CONTENIR LA CHAINE
0004 DD36FF20     00220         LD      (IX-1),20H  ;PAS DE SIGNE PAR DEFAUT
0008 CB7C         00230         BIT     7,H         ;LE NOMBRE EST-IL NEGATIF ?
000A 280B         00240         JR      Z,TRD       ;SI NON
000C 110000       00250         LD      DE,0        ;POUR SOUSTRAIRE HL DE 0
000F EB           00260         EX      DE,HL       ;
0010 B7           00270         OR      A           ;MISE DU CARRY A ZERO
0011 ED52         00280         SBC     HL,DE       ;FAIRE 0-HL POUR OBTENIR LA
                  00290                             ;VALEUR ABSOLUE DE HL
0013 DD36FF2D     00300         LD      (IX-1),'-'  ;METTRE UN SIGNE MOINS EN TETE
0017 111027       00310 TRD     LD      DE,10000    ;
001A CD3D00       00320         CALL    CAR         ;GARDER LE NOMBRE DES DIZAINES
                  00330                             ;DE MILLE
001D 11E803       00340         LD      DE,1000     ;
0020 CD3D00       00350         CALL    CAR         ;GARDER LE NOMBRE DES MILLES
0023 116400       00360         LD      DE,100      ;
0026 CD3D00       00370         CALL    CAR         ;GARDER LE NOMBRE DES CENTAINES
0029 110A00       00380         LD      DE,10       ;
002C CD3D00       00390         CALL    CAR         ;GARDER LE NOMBRE DES UNITES
002F 7D           00400         LD      A,L         ;L CONTIENT LE NOMBRE DES UNITES
0030 C630         00410         ADD     A,30H       ;TRANSFORMER EN ASCII
0032 DD7700       00420         LD      (IX+0),A    ;MISE DANS LE BUFFER A LA SUITE
0035 DD360100     00430         LD      (IX+1),0    ;METTRE UN ZERO A LA FIN
0039 214F00       00440         LD      HL,BUF      ;HL POINTE VERS LE PREMIER
                  00450                             ;ELEMENT
003C C9           00460         RET
                  00470 ;*****************************************************
                  00480 ;
                  00490 ; SOUS PROGRAMME DIVISANT HL PAR DE ET PLACANT LE
                  00500 ; QUOTIENT+30H DANS LE BUFFER
                  00510 ;
                  00520 ;*****************************************************
```

```
                00530 ;
003D B7         00540 CAR     OR      A         ;MISE A ZERO DU CARRY
003E 06FF       00550         LD      B,255     ;B=-1
0040 ED52       00560 G9      SBC     HL,DE     ;ON SOUSTRAIT HL PAR DE JUSQU'A
                00570                           ;CE QU'IL Y AIT DEBORDEMENT
0042 04         00580         INC     B         ;INCREMENTER LE QUOTIENT
0043 30FB       00590         JR      NC,G9     ;SI PAS DEBORDEMENT
0045 19         00600         ADD     HL,DE     ;HL CONTIENT LE RESTE
0046 78         00610         LD      A,B       ;A CONTIENT LE QUOTIENT
0047 C630       00620         ADD     A,30H     ;TRANSFORMATION EN ASCII
0049 DD7700     00630         LD      (IX+0),A  ;MISE DANS LE BUFFER
004C DD23       00640         INC     IX        ;INCREMENTER LE POINTEUR DU BUFFER
004E C9         00650         RET
0007            00660 BUF     DEFS    7         ;BUFFER CONTENANT LA CHAINE
0000            00670         END
00000 TOTAL ERRORS
```

2.7. LECTURE D'UNE LIGNE DE DONNÉES A PARTIR DU CLAVIER: CUR

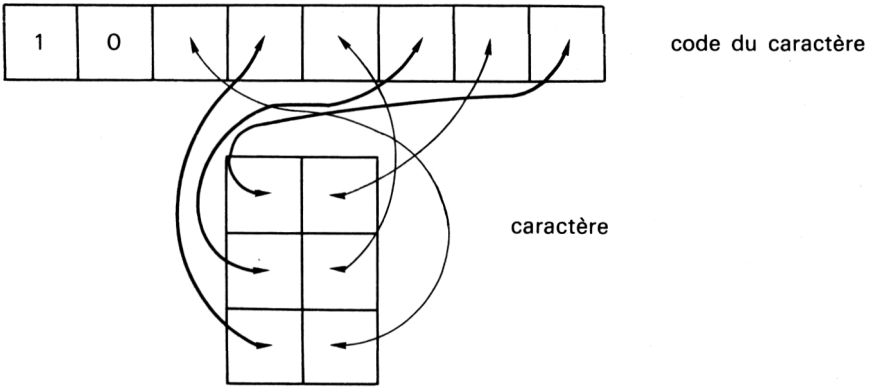
Cet utilitaire est destiné à permettre la saisie d'une suite de caractères ASCII tapés au clavier par le joueur. Il pourra être employé pour la lecture du nom du joueur en vue de l'affichage des scores, ou de la valeur d'un paramètre du jeu. L'entrée des données est gérée par un curseur clignotant répondant aux caractères de contrôle suivants : ENTER fin de la ligne ; ← effacement du dernier caractère ; SHIFT ← effacement de la ligne. Les caractères lus sont inscrits sur l'écran à partir de l'adresse donnée à l'appel du sous programme. Il faudra par conséquent régler le nombre maximal de caractères à lire (contenu dans C) de façon à ce qu'on ne dépasse pas la taille de la mémoire écran. Au retour du sous programme HL contient l'adresse du premier caractère de la chaine lue et E contient sa longueur. Il est alors aisé de l'exploiter en effectuant une lecture de la mémoire écran.

```
00010 ;*******************************************************
00020 ;
00030 ; LECTURE D'UNE LIGNE DE DONNEES A PARTIR DU CLAVIER,
00040 ; AVEC GESTION D'UN CURSEUR CLIGNOTANT
00050 ;
00060 ; EN ENTREE: HL CONTIENT L'ADRESSE A PARTIR DE LAQUELLE
00070 ;               S'INSCRIRONT LES CARACTERES TAPES PAR
00080 ;               L'UTILISATEUR
00090 ;            C  CONTIENT LE NOMBRE MAXIMAL DE CARACTERES
00100 ;
00110 ; EN SORTIE: HL POINTE VERS LA CHAINE ENTREE
00120 ;            E  CONTIENT LE NOMBRE DE CARACTERES LUS
00130 ;
00140 ;*******************************************************
```

```
                00150 ;
0000 368F       00160 CUR       LD      (HL),8FH;AFFICHAGE DU CURSEUR
0002 1E00       00170           LD      E,0     ;AUCUN CARACTERE LU
0004 E5         00180           PUSH    HL      ;SAUVEGARDER ADRESSE DE LA CHAINE
0005 0600       00190 AH        LD      B,0     ;COMPTEUR POUR LE CLIGNOTEMENT
0007 D5         00200 AD        PUSH    DE      ;SAUVEGARDER DE
0008 CD2B00     00210           CALL    2BH     ;LECTURE DU CLAVIER (CODE DU
                00220                           ;CARACTERE DANS A)
000B D1         00230           POP     DE      ;RESTAURER DE
000C B7         00240           OR      A       ;A=0 SI AUCUN CARACTERE N'A ETE
                00250                           ;FRAPPE
000D 2816       00260           JR      Z,AC    ;SI A=0
000F FE0D       00270           CP      0DH     ;EST-CE LE CODE DE ENTER ?
0011 2832       00280           JR      Z,AB    ;SI OUI
0013 FE08       00290           CP      8       ;EST-CE LA FLECHE GAUCHE ?
0015 2816       00300           JR      Z,AE    ;SI OUI LA TRAITER
0017 FE18       00310           CP      18H     ;EST-CE LA FLECHE GAUCHE + SHIFT ?
0019 281E       00320           JR      Z,AG    ;SI OUI TRAITER CE CAS
001B 57         00330           LD      D,A     ;SAUVEGARDER LE CARACTERE DANS D
001C 7B         00340           LD      A,E     ;NOMBRE DE CARACTERES
001D B9         00350           CP      C       ;EST-IL >= AU NOMBRE MAXIMAL ?
001E 3005       00360           JR      NC,AC   ;SI OUI NE RIEN FAIRE
0020 72         00370           LD      (HL),D  ;AFFICHER LE CARACTERE
0021 23         00380           INC     HL      ;INCREMENTER LE POINTEUR
0022 368F       00390           LD      (HL),8FH;REECRIRE LE CURSEUR
0024 1C         00400           INC     E       ;UN CARACTERE DE PLUS
0025 10E0       00410 AC        DJNZ    AD      ;DECREMENTER COMPTEUR DE
                00420                           ;CLIGNOTEMENT
0027 7E         00430           LD      A,(HL)  ;LECTURE DU CARACTERE EN BOUT
                00440                           ;DE LIGNE
0028 EE0F       00450           XOR     0FH     ;PASSER DE 80H A 8FH OU DE 8FH A 80H
002A 77         00460           LD      (HL),A  ;CLIGNOTEMENT
002B 18D8       00470           JR      AH      ;REINITIALISER LE COMPTEUR
002D 7B         00480 AE        LD      A,E     ;NOMBRE DE CARACTERES
002E B7         00490           OR      A       ;EST-IL NUL ?
002F 28F4       00500           JR      Z,AC    ;SI OUI NE RIEN EFFACER
0031 7E         00510           LD      A,(HL)  ;LECTURE DU CURSEUR
0032 3680       00520           LD      (HL),80H;EFFACER LE CURSEUR
0034 2B         00530           DEC     HL      ;DECREMENTER LE POINTEUR
0035 1D         00540           DEC     E       ;UN CARACTERE DE MOINS
0036 77         00550           LD      (HL),A  ;REECRIRE LE CURSEUR
0037 18EC       00560           JR      AC      ;CONTINUER
0039 7B         00570 AG        LD      A,E     ;NOMBRE DE CARACTERES
003A B7         00580           OR      A       ;EST-IL NUL ?
003B 28E8       00590           JR      Z,AC    ;SI OUI NE RIEN EFFACER
003D 7E         00600           LD      A,(HL)  ;LECTURE DU CURSEUR
003E 3680       00610           LD      (HL),80H;EFFACER LE CURSEUR
0040 2B         00620           DEC     HL      ;DECREMENTER LE POINTEUR
0041 1D         00630           DEC     E       ;UN CARACTERE DE MOINS
0042 77         00640           LD      (HL),A  ;REECRIRE LE CURSEUR
0043 18F4       00650           JR      AG      ;RECOMMENCER JUSQU'A CE QUE LA
                00660                           ;LIGNE SOIT VIDE
0045 E1         00670 AB        POP     HL      ;HL CONTIENT L'ADRESSE DE LA CHAINE
0046 C9         00680           RET
0000            00690           END
00000 TOTAL ERRORS
```

2.8. SOUS PROGRAMMES GRAPHIQUES : SET, RESET, POINT

Ces fonctions utilisent la structure interne du graphisme de la mémoire écran. Rappelons à ce propos qu'un caractère graphique est constitué de six segments, chacun de ces segments correspond à un des bits du code du caractère selon le schéma suivant :



Ces trois sous programmes reçoivent l'abscisse et l'ordonnée du segment graphique. Ils appellent alors tous le sous programme H9 qui détermine la case mémoire correspondante et le code graphique à y mettre pour obtenir le segment seul. Ceci se fait en employant des opérations arithmétiques (division, multiplication...) pour calculer l'expression : adresse = 3COOH + 64 * INT (Y/3) + INT (X/2) ; où X et Y désignent respectivement l'abcisse et l'ordonnée du segment graphique. Ensuite ces informations sont exploitées différemment par les sous programmes afin de produire la fonction désirée.

Lors de l'appel, il faudra s'assurer que les valeurs contenues dans D et E correspondent bien à l'intervalle spécifié, sinon l'affichage se fera en dehors de la mémoire écran, ce qui pourrait avoir des conséquences facheuses ! Il faudra également avoir initialisé l'écran en mode graphique par le programme CLS qui remplit l'écran de caractères 80 H.

```
                00010 ;*******************************************************
                00020 ;
                00030 ; FONCTIONS GRAPHIQUES SET, RESET, POINT ANALOGUES A
                00040 ; CELLES UTILISEES EN BASIC
                00050 ;
                00060 ; EN ENTREE: D  CONTIENT L'ORDONNEE DU SEGMENT GRAPHIQUE
                00070 ;               0 <= D <=47
                00080 ;            E  CONTIENT L'ABCISSE DU SEGMENT GRAPHIQUE
                00090 ;               0 <= E <= 127
                00100 ;
                00110 ; TOUS LES REGISTRES SONT SAUVEGARDES SAUF AF
                00120 ;
                00130 ;*******************************************************
                00140 ;
                00150 ;=======================================================
                00160 ;
                00170 ; ALLUMAGE D'UN SEGMENT GRAPHIQUE
                00180 ;
                00190 ;=======================================================
                00200 ;
0000 E5         00210 SET     PUSH    HL        ;SAUVEGARDER HL
0001 CD1800     00220         CALL    H9        ;RECHERCHE DE LA CASE MEMOIRE ECRAN
0004 B6         00230         OR      (HL)      ;MISE A 1 DU BIT CORRESPONDANT
0005 77         00240         LD      (HL),A    ;AFFICHAGE DU SEGMENT
0006 E1         00250         POP     HL        ;RESTAURER HL
0007 C9         00260         RET
                00270 ;=======================================================
                00280 ;
                00290 ; EXTINCTION D'UN SEGMENT GRAPHIQUE
                00300 ;
                00310 ;=======================================================
                00320 ;
0008 E5         00330 RESET   PUSH    HL        ;SAUVEGARDER HL
0009 CD1800     00340         CALL    H9        ;RECHERCHE DE LA CASE MEMOIRE ECRAN
000C 2F         00350         CPL               ;TOUS LES BITS DE A SONT A 1 SAUF 1
000D A6         00360         AND     (HL)      ;MISE A 0 DU BIT CORRESPONDANT
000E 77         00370         LD      (HL),A    ;EXTINCTION DU SEGMENT
000F E1         00380         POP     HL        ;RESTAURER HL
0010 C9         00390         RET
                00400 ;=======================================================
                00410 ;
                00420 ; TEST DE L'ALLUMAGE D'UN SEGMENT GRAPHIQUE
                00430 ; LE FLAG Z EST A 1 SI LE SEGMENT EST ETEINT
                00440 ;           EST A 0 SI LE SEGMENT EST ALLUME
                00450 ;
                00460 ;=======================================================
                00470 ;
0011 E5         00480 POINT   PUSH    HL        ;SAUVEGARDER HL
0012 CD1800     00490         CALL    H9        ;RECHERCHE CASE MEMOIRE ECRAN
0015 A6         00500         AND     (HL)      ;TEST DU BIT CORRESPONDANT
0016 E1         00510         POP     HL        ;RESTAURER HL
0017 C9         00520         RET
                00530 ;=======================================================
```

```
                00540 ;
                00550 ; SOUS PROGRAMME DETERMINANT LA CASE MEMOIRE DE L'ECRAN
                00560 ; CORRESPONDANT A UN SEGMENT GRAPHIQUE
                00570 ;
                00580 ; EN SORTIE: HL CONTIENT LA CASE MEMOIRE ECRAN
                00590 ;            A  CONTIENT LE CODE GRAPHIQUE CORRESPONDANT
                00600 ;               AU SEGMENT SEUL
                00610 ;
                00620 ;=================================================
                00630 ;
0018 D5         00640 H9      PUSH    DE        ;SAUVEGARDER REGISTRES DE
0019 C5         00650         PUSH    BC        ;ET BC
001A 7A         00660         LD      A,D       ;DIVISER D PAR 3 POUR OBTENIR LE
                00670                           ;NUMERO DE LIGNE (MIS DANS L)
001B 2EFF       00680         LD      L,0FFH    ;L=-1
001D 2C         00690 H5      INC     L         ;INCREMENTER LE QUOTIENT
001E D603       00700         SUB     3         ;SOUSTRAIRE 3 TANT QU'IL N'Y A
                00710                           ;PAS DEBORDEMENT
0020 F21D00     00720         JP      P,H5      ;SI PAS DEBORDEMENT
0023 C603       00730         ADD     A,3       ;A CONTIENT LE RESTE DE LA DIVISION
0025 CB27       00740         SLA     A         ;MULTIPLIER PAR 2
0027 3C         00750         INC     A         ;+1
0028 47         00760         LD      B,A       ;NOMBRE DE DECALAGES A EFFECTUER
0029 3E80       00770         LD      A,80H     ;BIT 7 A 1
002B 07         00780 H6      RLCA              ;DECALAGE POUR OBTENIR LE CODE
                00790                           ;GRAPHIQUE CORRESPONDANT AU SEGMENT
002C 10FD       00800         DJNZ    H6        ;B DECALAGES
002E 2600       00810         LD      H,0       ;HL CONTIENT LE NUMERO DE LIGNE
0030 0606       00820         LD      B,6       ;COMPTEUR DE BOUCLE
0032 29         00830 H7      ADD     HL,HL     ;MULTIPLICATION PAR 64
0033 10FD       00840         DJNZ    H7        ;
0035 163C       00850         LD      D,3CH     ;D CONTIENT L'OCTET DE POIDS FAIBLE
                00860                           ;DE L'ADRESSE DE LA MEMOIRE ECRAN
0037 CB3B       00870         SRL     E         ;DIVISION PAR 2, E CONTIENT LA
                00880                           ;POSITION DU CARACTERE DANS LA LIGNE
0039 3001       00890         JR      NC,H8     ;SI E ETAIT PAIR
003B 07         00900         RLCA              ;SINON MODIFIER LE CODE
003C 19         00910 H8      ADD     HL,DE     ;HL CONTIENT L'ADRESSE DU CARACTERE
003D C1         00920         POP     BC        ;RESTAURER LES REGISTRES BC
003E D1         00930         POP     DE        ;ET DE
003F C9         00940         RET
0000            00950         END
00000 TOTAL ERRORS
```

2.9. TRACÉ DE DROITE: DROITE

Ce programme permet le tracé sur l'écran de n'importe quelle droite définie par ses coordonnées initiales (X_1 et Y_1) et ses coordonnées finales (X_2 et Y_2). Pour expliquer l'algorithme employé nous supposerons que la variation en Y est inférieure à la variation en X, soit:

$$|Y_2 - Y_1| < |X_2 - X_1| \text{ et que } X_1 < X_2;\ Y_1 < Y_2$$

On utilise une boucle pour pouvoir dessiner successivement tous les points de la droite. L'abscisse du point courant est incrémentée à chaque passage. Au contraire l'ordonnée est incrémentée chaque fois que cela est nécessaire pour conserver la direction de la droite. Tout le problème consiste à savoir quand faut-il l'incrémenter. Pour cela examinons l'équation de la droite :

$$Y = Y_1 + (X - X_1)\frac{Y_2 - Y_1}{X_2 - X_1}$$

Soit X_i l'abscisse du point courant de la droite et X_{i+1} l'abscisse suivante.

On a : $X_{i+1} = X_i + 1$

et $$Y_{i+1} = Y_1 + (X_i + 1 - X_1)\frac{Y_2 - Y_1}{X_2 - X_1}$$

$$Y_{i+1} = Y_i + \frac{Y_2 - Y_1}{X_2 - X_1}$$

Entre deux points l'ordonnée est augmentée de $\frac{Y_2 - Y_1}{X_2 - X_1}$ qui est inférieur à un d'après notre hypothèse. Si on néglige ce terme en n'augmentant pas Y on commet une erreur de $\frac{Y_2 - Y_1}{X_2 - X_1}$. Aussi au point suivant on aura : $Y_i + 2 = Y_i + 2 * \frac{Y_2 - Y_1}{X_2 - X_1}$.

Si on n'augmente pas Y pendant n points l'erreur cumulée sera égale à $n * \frac{Y_2 - Y_1}{X_2 - X_1}$. Quand cette valeur deviendra supérieure à 0,5 il faudra incrémenter l'ordonnée. La procédure utilisée dans le programme découle de ce que nous venons de dire. On gère une variable noté ERR qui contient non pas l'erreur proprement dite qui est un nombre fractionnaire, mais un nombre entier égal à : $n * (Y_2 - Y_1) + \frac{X_2 - X_1}{2}$. L'erreur sera supérieure à 0,5 si ce nombre dépasse $X_2 - X_1$.

$$n * (Y_2 - Y_1) + \frac{X_2 - X_1}{2} > X_2 - X_1 \Rightarrow n * \frac{(Y_2 - Y_1)}{X_2 - X_1} > 0{,}5$$

Il faudra alors incrémenter l'ordonnée, et soustraire la valeur $X_2 - X_1$ à la variable ERR, ce qui revient à soustraire 1 de l'erreur. On obtient ainsi le tracé complet de la droite en faisant varier X_i de X_1 à X_2.

Le programme général contient une phase d'initialisation destinée à calculer $|X_2 - X_1|$ et $|Y_2 - Y_1|$ et à déterminer l'incrément à ajouter à X et à Y (– 1 ou + 1) suivant le signe de $X_2 - X_1$ et de $Y_2 - Y_1$. On trouve ensuite deux boucles pour le tracé de la droite correspondant à chacun des cas: $|X_2 - X_1| > |Y_2 - Y_1|$ et $|X_2 - X_1| < |Y_2 - Y_1|$. Il est alors possible de tracer n'importe quelle droite pourvu que ses coordonnées initiales et finales soient comprises dans les limites admises et qu'elles soient différentes. Comme pour les fonctions graphiques, il faudra, avant de tracer la droite, effacer l'écran par le programme CLS remplissant celui-ci de caractères de code 80 H.

REMARQUE : Il est possible de se servir de ce sous programme pour effacer une droite en remplaçant SET par RESET.)

```
                00010 ;*****************************************************
                00020 ;
                00030 ; TRACEUR DE DROITE SUR L'ECRAN
                00040 ;
                00050 ; EN ENTREE: D  CONTIENT L'ORDONNEE INITIALE Y1
                00060 ;            E  CONTIENT L'ABCISSE INITIALE X1
                00070 ;            H  CONTIENT L'ORDONNEE FINALE Y2
                00080 ;            L  CONTIENT L'ABCISSE FINALE X2
                00090 ;
                00100 ;*****************************************************
                00110 ;
0000 0601       00120 DROITE  LD      B,1       ;L'INCREMENT EN Y VAUT 1 PAR DEFAUT
0002 48         00130         LD      C,B       ;L'INCREMENT EN X VAUT 1 PAR DEFAUT
0003 7C         00140         LD      A,H       ;A=Y2
0004 92         00150         SUB     D         ;A=Y2-Y1
0005 3004       00160         JR      NC,G0     ;SI Y2-Y1>=0 NE RIEN FAIRE
0007 06FF       00170         LD      B,0FFH    ;SINON PRENDRE -1 COMME INCREMENT
0009 ED44       00180         NEG               ;A=-(Y2-Y1)
000B 67         00190 G0      LD      H,A       ;H=ABS(Y2-Y1)
000C 7D         00200         LD      A,L       ;A=X2
000D 93         00210         SUB     E         ;A=X2-X1
000E 3004       00220         JR      NC,G1     ;SI X2-X1>=0 NE RIEN FAIRE
0010 0EFF       00230         LD      C,0FFH    ;SINON PRENDRE -1 COMME INCREMENT
0012 ED44       00240         NEG               ;A=-(X2-X1)
0014 6F         00250 G1      LD      L,A       ;L=ABS(X2-X1)
0015 ED435700   00260         LD      (G3),BC   ;SAUVEGARDER LES INCREMENTS DANS G3
0019 7C         00270         LD      A,H       ;A=ABS(Y2-Y1)
001A BD         00280         CP      L         ;EST-CE QUE ABS(Y2-Y1)>=ABS(X2-X1) ?
001B 301C       00290         JR      NC,G2     ;SI OUI PASSER A LA SECONDE BOUCLE
001D 45         00300         LD      B,L       ;B=ABS(X2-X1) : COMPTEUR DE BOUCLE
001E 4D         00310         LD      C,L       ;C=ERR
001F CB39       00320         SRL     C         ;ERR=ABS(X2-X1)/2
0021 CD5900     00330 G5      CALL    SET       ;AFFICHAGE DU SEGMENT D,E
```

```
0024 3A5700    00340          LD      A,(G3)  ;INCREMENT EN X
0027 83        00350          ADD     A,E     ;AJOUTER L'INCREMENT A X
0028 5F        00360          LD      E,A     ;NOUVEL X
0029 79        00370          LD      A,C     ;A=ERR
002A 84        00380          ADD     A,H     ;A=ERR+ABS(Y2-Y1)
002B 4F        00390          LD      C,A     ;ERR=ERR+ABS(Y2-Y1)
002C 95        00400          SUB     L       ;SOUSTRAIRE ABS(X2-X1)
002D 3806      00410          JR      C,G4    ;SI ERR<ABS(X2-X1) PASSER AU POINT
               00420                          ;SUIVANT
002F 4F        00430          LD      C,A     ;ERR=ERR-ABS(X2-X1)
0030 3A5800    00440          LD      A,(G3+1);INCREMENT EN Y
0033 82        00450          ADD     A,D     ;INCREMENTER Y
0034 57        00460          LD      D,A     ;NOUVEL Y
0035 10EA      00470 G4       DJNZ    G5      ;BOUCLE POUR AFFICHER TOUS LES
               00480                          ;POINTS
0037 181A      00490          JR      G6      ;AFFICHER LE POINT FINAL
0039 44        00500 G2       LD      B,H     ;B=ABS(Y2-Y1) : COMPTEUR DE BOUCLE
003A 4C        00510          LD      C,H     ;ERR=C
003B CB39      00520          SRL     C       ;ERR=ABS(Y2-Y1)/2
003D CD5900    00530 G7       CALL    SET     ;AFFICHAGE D'UN SEGMENT
0040 3A5800    00540          LD      A,(G3+1);INCREMENT EN Y
0043 82        00550          ADD     A,D     ;INCREMENTER Y
0044 57        00560          LD      D,A     ;NOUVEL Y
0045 79        00570          LD      A,C     ;A=ERR
0046 85        00580          ADD     A,L     ;A=ERR+ABS(X2-X1)
0047 4F        00590          LD      C,A     ;ERR=A
0048 94        00600          SUB     H       ;SOUTRAIRE ABS(Y2-Y1)
0049 3806      00610          JR      C,G8    ;SI ERR<ABS(Y2-Y1) PASSER AU POINT
               00620                          ;SUIVANT
004B 4F        00630          LD      C,A     ;ERR=ERR-ABS(Y2-Y1)
004C 3A5700    00640          LD      A,(G3)  ;INCREMENT EN X
004F 83        00650          ADD     A,E     ;INCREMENTER X
0050 5F        00660          LD      E,A     ;NOUVEL X
0051 10EA      00670 G8       DJNZ    G7      ;BOUCLE POUR AFFICHER TOUS LES
               00680                          ;POINTS
0053 CD5900    00690 G6       CALL    SET     ;AFFICHAGE DU DERNIER POINT
0056 C9        00700          RET
0002           00710 G3       DEFS    2
0059           00720 SET      EQU     $       ;LE SOUS PROGRAMME SET DEVRA ETRE
               00730                          ;IMPLANTE ICI
0000           00740          END
00000 TOTAL ERRORS
```

2.10. SOUS PROGRAMME DONNANT LE NUMÉRO DE LIGNE CORRESPONDANT A UNE CASE MÉMOIRE : LIGNE

Ce sous programme est très simple à comprendre. Il exploite seulement le fait que la mémoire écran débute en 3COOH et que les lignes possèdent 64 caractères. Étant donné qu'il risque d'être appelé souvent dans un programme de jeu, nous avons préféré le donner dans ce chapitre pour des raisons de standardisation.

```
                00010 ;*****************************************************
                00020 ;
                00030 ; SOUS PROGRAMME DONNANT LE NUMERO DE LIGNE D'UNE CASE
                00040 ; MEMOIRE DONT ON CONNAIT L'ADRESSE
                00050 ;
                00060 ; EN ENTREE: HL CONTIENT L'ADRESSE DE LA CASE MEMOIRE
                00070 ;
                00080 ; EN SORTIE: A  CONTIENT LE NUMERO DE LIGNE
                00090 ;               COMPRIS ENTRE 0 ET 15
                00100 ;
                00110 ; TOUS LES REGISTRES SAUF AF SONT SAUVEGARDES
                00120 ;
                00130 ;*****************************************************
                00140 ;
0000 D5         00150 LIGNE   PUSH    DE        ;SAUVEGARDER DE
0001 E5         00155         PUSH    HL        ;ET HL
0002 11003C     00160         LD      DE,3C00H  ;DEBUT DE LA MEMOIRE ECRAN
0005 B7         00170         OR      A         ;MISE A ZERO DU CARRY
0006 ED52       00180         SBC     HL,DE     ;HL CONTIENT L'ADRESSE RELATIVE DE
                00190                           ;LA CASE PAR RAPPORT AU DEBUT DE
                00200                           ;LA MEMOIRE ECRAN
0008 29         00210         ADD     HL,HL     ;MULTIPLIER CETTE ADRESSE PAR 2
0009 29         00220         ADD     HL,HL     ;PAR 4
000A 7C         00230         LD      A,H       ;A CONTIENT LE NUMERO DE LIGNE
                00240                           ;( 0 <= A <= 15 )
000B E1         00250         POP     HL        ;RESTAURER HL
000C D1         00255         POP     DE        ;ET DE
000D C9         00260         RET
0000            00270         END
00000 TOTAL ERRORS
```

2.11. SOUS PROGRAMMES DIVERS IMPLANTÉS DANS LA ROM

Nous allons décrire dans ce chapitre certains sous programmes particulièrement utiles, qui existent aussi bien sur le modèle 1 que sur le modèle 3. Il serait par conséquent inutile de les dupliquer dans votre programme.

2.11.1. RST 18 H

Ce sous programme peut être appelé par une instruction RST du Z 80 qui occupe un octet, soit deux octets de moins qu'une instruction CALL. Vous ne devrez donc avoir aucun scrupule à l'utiliser. Il effectue une comparaison entre les deux registres de 16 bits HL et DE. A la sortie du sous-programme le flag Z vaut 1 si DE = HL, le flag carry vaut 1 si DE > HL. Par ailleurs tous les registres sont sauvegardés sauf AF.

2.11.2. CALL 2BH

Ce sous programme effectue une lecture du clavier et fournit dans A le code du caractère qui a été frappé, A vaut zéro si aucun caractère n'a été modifié. Tous les registres sont sauvegardés sauf AF et DE. Ce programme sera utile chaque fois que l'on voudra lire une donnée sur le clavier. Toutefois il pourra s'avérer être trop lent dans les parties les plus mouvementées de votre programme de jeu. On aura alors recours à une lecture directe de la mémoire clavier.

2.11.3. CALL 60 H

Voici un petit utilitaire destiné à provoquer un temps d'attente programmable. Avant l'appel du sous programme mettez une valeur proportionnelle au temps d'attente dans le registre BC. La durée du sous programme sera égale à : (BC) * 14,65 ns. Si BC est nul le temps d'attente sera proportionnel à 65536. Seuls les registres BC et AF sont détruits par le sous programme.

3

Graphismes

Voici, vous vous en doutez, la portion la plus importante de la réalisation d'un programme de jeu. En effet, la plupart des parties du jeu résultent de dessins ou de déplacements de graphismes, que ce soit pour déplacer les ennemis, faire avancer une balle ou bouger le vaisseau du joueur. Nous n'avons pas la prétention dans ce livre de vous énumérer toutes les façons de produire des graphismes mobiles sur le TRS 80. Il y en a de nombreuses qui ont chacune des points communs. Nous exposerons celles que nous connaissons et qui ont déjà fait leurs preuves. Elles devraient suffire à réaliser n'importe quel type de jeu.

3.1. STRUCTURE DE L'ÉCRAN

Pour faciliter les opérations d'affichage, le TRS 80 utilise une mémoire vive de 1 koctet qui sert d'interface entre l'écran et l'unité centrale. Cette mémoire est lue de façon cyclique par un procédé matériel générant les signaux nécessaires à l'affichage des caractères qui y sont contenus. La mémoire écran est implantée entre les adresses 3COOH (15360) et 3FFFH (16383). Elle est divisée en seize lignes de 64 octets chacune. La première ligne située en haut de l'écran correspond aux adresses 3COOH à 3C3FH (de gauche à droite), la dernière ligne située en bas de l'écran utilise les adresses 3FCOH à 3FFFH. Ainsi pour affi-

cher un caractère sur la ligne L à la position P il faut inscrire son code à l'adresse : 3COOH + 64 * L + P.

Les caractères graphiques sont constitués de six segments distincts qui sont soit allumés soit éteints. Chaque segment allumé correspond à un bit à 1 du code du caractère selon le schéma donné dans le chapitre 2 à propos du sous programme SET. Il est possible d'exploiter cette structure pour faire un affichage par segment ou lieu d'un affichage par caractère, ce qui autorise une plus grande souplesse d'utilisation.

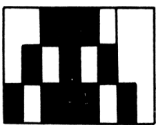
3.2. DESSIN DES GRAPHISMES SUR L'ÉCRAN

3.2.1. Affichage par caractère

Un personnage du jeu est constitué d'un groupe de caractères graphiques plus ou moins large, implanté dans un certain endroit de l'écran. Pour le dessiner, il suffira donc de mettre dans les bonnes cases de la mémoire écran les bons codes de caractère. Il est conseillé de commencer à le tracer sur une feuille de papier quadrillée de façon à reproduire la division de l'écran en segments et en caractères. Ensuite vous pourrez déterminer les codes graphiques par comparaison entre votre dessin et la table des caractères graphiques donnée en annexe. Si votre personnage tient sur une seule ligne, il pourra être affiché par le sous programme MES du chapitre 2 en employant une suite de caractères graphiques le représentant et terminée par un zéro. Le registre DE sera utilisé pour indiquer l'adresse à partir de laquelle il doit être implanté.

Exemple :

On désire afficher le dessin suivant :



Codes des caractères :
(en hexadécimal)

98,BB,99,90

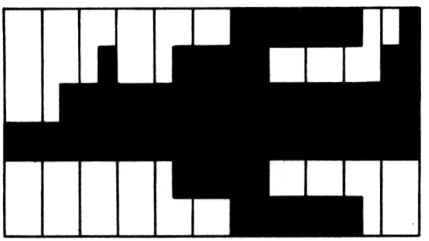
Le programme suivant affichera ce dessin :

```
7A00            00010           ORG     7A00H
7A00 119E3D     00020 DEBUT     LD      DE,3D9EH ;ADRESSE OU DOIT ETRE IMPLANTE
                00030                            ;LE DESSIN
7A03 21137A     00040           LD      HL,DESSIN;ADRESSE DE LA CHAINE GRAPHIQUE
7A06 CD0B7A     00050           CALL    MES      ;AFFICHAGE DU PERSONNAGE
7A09 18FE       00060 LA        JR      LA       ;ARRET
7A0B 7E         00070 MES       LD      A,(HL)   ;SOUS PROGRAMME D'AFFICHAGE
7A0C B7         00080           OR      A        ;(VOIR CHAPITRE 2)
7A0D C8         00090           RET     Z        ;
7A0E 12         00100           LD      (DE),A   ;
7A0F 13         00110           INC     DE       ;
7A10 23         00120           INC     HL       ;
7A11 18F8       00130           JR      MES      ;
7A13 98         00140 DESSIN    DEFB    98H      ;SUITE DE CARACTERES GRAPHIQUES
7A14 BB         00150           DEFB    0BBH     ;REPRESENTANT LE PERSONNAGE
7A15 99         00160           DEFB    99H      ;
7A16 90         00170           DEFB    90H      ;
7A17 00         00180           DEFB    0        ;TERMINEE PAR UN ZERO
7A00            00190           END     DEBUT
00000 TOTAL ERRORS
```

Si le personnage occupe plus d'une ligne, on utilisera un autre sous programme noté AFF employant une structure de chaîne analogue. Un code de caractère spécial (par exemple 1) indiquera le passage à la ligne suivante. Le personnage sera décrit par une suite de caractères graphiques terminée par un zéro et comportant éventuellement des 1 pour indiquer un saut de ligne. L'appel du sous programme sera strictement identique à celui de MES, DE contenant l'adresse d'implantation du caractère le plus en haut et à gauche.

Exemple :

Affichage d'un avion :



Code des caractères : 80,A0,B8,B0,B8,BC,BF,B3,B3,B1,BE 1re ligne
(en hexadécimal)

83,83,83,83,8B,8F,BF,B3,B3,93,83 2e ligne

Le programme suivant permet d'afficher cet avion.

```
7A00            00010           ORG     7A00H
7A00 119A3D     00020 DEBUT     LD      DE,3D9AH ;ADRESSE OU DOIT ETRE IMPLANTE
                00030                            ;LE DESSIN
7A03 21247A     00040           LD      HL,DESSIN;ADRESSE DE LA CHAINE GRAPHIQUE
7A06 CD0B7A     00050           CALL    AFF      ;AFFICHAGE DE L'AVION
7A09 18FE       00060 LA        JR      LA       ;ARRET
                00070 ;
                00080 ;SOUS PROGRAMME D'AFFICHAGE
                00090 ;
7A0B D5         00100 AFF       PUSH    DE       ;SAUVEGARDE ADR. DEBUT DE LIGNE
7A0C 7E         00110 LOOP      LD      A,(HL)   ;CARACTERE DANS A
7A0D 23         00120           INC     HL       ;INCREMENTER POINTEUR SUR CHAINE
7A0E B7         00130           OR      A        ;CARACTERE NUL ?
7A0F 2811       00140           JR      Z,FIN    ;SI NUL FIN DU SOUS PROGRAMME
7A11 FE01       00150           CP      1        ;EST-CE UN SAUT DE LIGNE ?
7A13 2804       00160           JR      Z,SAUT   ;SI OUI FAIRE UN SAUT DE LIGNE
7A15 12         00170           LD      (DE),A   ;AFFICHAGE D'UN CARACTERE
7A16 13         00180           INC     DE       ;INCREMENTER POINTEUR SUR ECRAN
7A17 18F3       00190           JR      LOOP     ;PASSER AU CARACTERE SUIVANT
7A19 E3         00200 SAUT      EX      (SP),HL  ;SAUVEGARDER HL EN PILE ET METTRE
                00210                            ;L'ADR. DE DEBUT DE LIGNE DANS HL
7A1A 114000     00220           LD      DE,64    ;NOMBRE DE CARACTERES PAR LIGNE
7A1D 19         00230           ADD     HL,DE    ;HL POINTE VERS LA LIGNE SUIVANTE
7A1E EB         00240           EX      DE,HL    ;DE POINTE VERS LE LIGNE SUIVANTE
7A1F E1         00250           POP     HL       ;RESTAURER POINTEUR SUR CHAINE
7A20 18E9       00260           JR      AFF      ;PASSER AU CARACTERE SUIVANT
7A22 D1         00270 FIN       POP     DE       ;DEPILER DE
7A23 C9         00280           RET              ;RETOUR
                00290 ;
                00300 ;CHAINE GRAPHIQUE REPRESENTANT L'AVION
                00310 ;LES OCTETS SONT GROUPES PAR 2 EN UTILISANT LA DIRECTIVE
                00320 ;DEFW POUR DIMINUER LE NOMBRE DE LIGNES DU PROGRAMME
                00330 ;
7A24 80A0       00340 DESSIN    DEFW    0A080H   ;PREMIERE LIGNE
7A26 B8B0       00350           DEFW    0B0B8H
7A28 B8BC       00360           DEFW    0BCB8H
7A2A BFB3       00370           DEFW    0B3BFH
7A2C B3B1       00380           DEFW    0B1B3H
7A2E BE01       00390           DEFW    1BEH     ;SAUT DE LIGNE
7A30 8383       00400           DEFW    8383H    ;SECONDE LIGNE
7A32 8383       00410           DEFW    8383H
7A34 8F8F       00420           DEFW    8F8FH
7A36 BFB3       00430           DEFW    0B3BFH
7A38 B393       00440           DEFW    93B3H
7A3A 8300       00450           DEFW    83H      ;FIN DE LA CHAINE
7A00            00460           END     DEBUT
00000 TOTAL ERRORS
```

La méthode décrite dans ce paragraphe possède l'avantage d'être très rapide et de pouvoir facilement s'adapter à une extension haute résolution utilisant un générateur de caractères graphiques (80 grafix). Elle peut être utilisée dans de nombreux cas, parfois au prix de quelques limitations. Nous utiliserons essentiellement cette méthode dans le programme de jeu SONAR.

3.2.2. Affichage exploitant la structure du code des caractères

Au lieu d'afficher un caractère graphique en inscrivant tout bonnement son code dans une case mémoire de l'écran, il est possible de le superposer au caractère qui s'y trouvait précédemment en utilisant les fonctions logiques du Z 80. On pourra ainsi dessiner indépendemment deux personnages voisins qui partagent une ou plusieurs cases mémoire. Parallèlement à cet affichage on peut créer un sous programme qui efface un personnage sans effacer un seul des autres segments de l'écran, et un autre qui teste si un des segments du personnage va se placer sur un segment déjà allumé de l'écran. On obtient ainsi quasiment un affichage par segment en conservant la rapidité de l'affichage par caractère. Pour arriver à ce but, il suffit de remplacer dans le programme AFF du paragraphe précédent l'instruction LD (DE), A (Ligne 170) par un groupe d'instructions variant suivant l'application désirée (affichage, effacement, test).

```
00170 ;
00171 ;AFFICHAGE PAR BIT DU CARACTERE
00172 ;
00173          LD       B,A        ;SAUVEGARDE CARACTERE DANS B
00174          LD       A,(DE)     ;LECTURE CARACTERE PRECEDENT
00175          OR       B          ;MIXER LES DEUX CODES GRAPHIQUES
00176          LD       (DE),A     ;AFFICHER LE NOUVEAU CARACTERE
```

```
00170 ;
00171 ;EFFACEMENT PAR BIT DU CARACTERE
00172 ;
00173          XOR      3FH        ;TOUS LES BITS A 1 DU CODE SONT
00174                              ;MIS A 0 ET CEUX A 0 SONT MIS A 1
00175          LD       B,A        ;SAUVEGARDE CARACTERE INVERSE
00176          LD       A,(DE)     ;LECTURE CARACTERE PRECEDENT
00177          AND      B          ;MIXER LES DEUX CODES GRAPHIQUES
00178          LD       (DE),A     ;AFFICHER LE NOUVEAU CARACTERE
```

```
00170 ;
00171 ;TEST PAR BIT DU CARACTERE
00172 ;
00173          LD       B,A        ;SAUVEGARDE CARACTERE DANS B
00174          LD       A,(DE)     ;LECTURE CARACTERE PRECEDENT
00175          AND      B          ;TESTER BIT PAR BIT
00176          AND      3FH        ;EFFACER LES BITS 6 ET 7
00177                              ;Z=1 SI LES SEGMENTS DU NOUVEAU
00178                              ;CARACTERE SONT DIFFERENTS DES
00179                              ;SEGMENTS DE L'ANCIEN
```

Il est possible de regrouper ces trois fonctions en un seul sous programme noté AFBIT. Lors de l'appel DE et HL seront initialisés de la

même façon que pour le programme AFF, et de plus le registre C indiquera le type de fonction désiré:

C = 0 affichage, C = 1 effacement, C = 2 test.

Le listing ci-dessous contient le sous programme AFBIT auquel nous avons ajouté un petit programme montrant son fonctionnement. Ce programme affiche un caractère au milieu de l'écran, puis, après que l'utilisateur ait appuyé sur une touche, il appelle AFBIT pour dessiner un vaisseau par-dessus. Ensuite il fait de même pour l'effacement du vaisseau.

```
7D00            00010           ORG     7D00H
7D00 21003C     00020 DEBUT     LD      HL,3C00H ;EFFACEMENT DE L'ECRAN
7D03 11013C     00030           LD      DE,3C01H ;
7D06 3680       00040           LD      (HL),80H ;
7D08 01FF03     00050           LD      BC,3FFH  ;
7D0B EDB0       00060           LDIR             ;
7D0D 3E8C       00070           LD      A,8CH    ;CARACTERE GRAPHIQUE
7D0F 32A03D     00080           LD      (3DA0H),A;CELUI-CI EST AFFICHE
7D12 CD4900     00090           CALL    49H      ;ATTENTE FRAPPE D'UNE TOUCHE
7D15 119E3D     00100           LD      DE,3D9EH ;ADRESSE OU COMMENCE LE DESSIN
7D18 21657D     00110           LD      HL,DESSIN;ADRESSE DE LA CHAINE GRAPHIQUE
7D1B 0E00       00120           LD      C,0      ;CODE AFFICHAGE
7D1D CD307D     00130           CALL    AFBIT    ;AFFICHAGE
7D20 CD4900     00140           CALL    49H      ;ATTENTE FRAPPE D'UNE TOUCHE
7D23 119E3D     00150           LD      DE,3D9EH ;ADRESSE OU COMMENCE LE DESSIN
7D26 21657D     00160           LD      HL,DESSIN;ADRESSE DE LA CHAINE GRAPHIQUE
7D29 0E01       00170           LD      C,1      ;CODE EFFACEMENT
7D2B CD307D     00180           CALL    AFBIT    ;EFFACEMENT
7D2E 18FE       00190 LA        JR      LA       ;ARRET
                00200 ;*****************************************************
                00210 ;
                00220 ;SOUS PROGRAMME D'AFFICHAGE PAR BIT
                00230 ;
                00240 ;EN ENTREE: HL=ADRESSE DE LA CHAINE GRAPHIQUE
                00250 ;           DE=ADRESSE OU COMMENCE LA CHAINE
                00260 ;           C=0 --> AFFICHAGE
                00270 ;           C=1 --> EFFACEMENT
                00280 ;           C=2 --> TEST
                00290 ;
                00300 ;*****************************************************
7D30 D5         00310 AFBIT     PUSH    DE       ;SAUVEGARDE ADR. DEBUT DE LIGNE
7D31 7E         00320 LOOP      LD      A,(HL)   ;CARACTERE DANS A
7D32 23         00330           INC     HL       ;INCREMENTER POINTEUR SUR CHAINE
7D33 B7         00340           OR      A        ;CARACTERE NUL ?
7D34 282D       00350           JR      Z,FIN    ;SI NUL FIN DU SOUS PROGRAMME
7D36 FE01       00360           CP      1        ;EST-CE UN SAUT DE LIGNE ?
7D38 2820       00370           JR      Z,SAUT   ;SI OUI FAIRE UN SAUT DE LIGNE
7D3A CB41       00380           BIT     0,C      ;EST-CE LE CODE 1 ?
7D3C 200A       00390           JR      NZ,EFFA  ;SI OUI EFFACEMENT
7D3E CB49       00400           BIT     1,C      ;EST-CE LE CODE 2 ?
7D40 200E       00410           JR      NZ,TEST  ;SI OUI TEST
                00420 ;AFFICHAGE
7D42 47         00430 AFFI      LD      B,A      ;SAUVEGARDE CARACTERE DANS B
7D43 1A         00440           LD      A,(DE)   ;LECTURE CARACTERE PRECEDENT
7D44 B0         00450           OR      B        ;MIXER LES 2 CODES GRAPHIQUES
7D45 12         00460           LD      (DE),A   ;AFFICHER LE NOUVEAU CARACTERE
7D46 180F       00470           JR      ICI      ;PASSER AU CARACTERE SUIVANT
                00480 ;EFFACEMENT
```

```
7D48 EE3F      00490 EFFA   XOR     3FH       ;TOUS LES BITS A 1 DU CODE SONT
               00500                          ;MIS A 0 ET CEUX A 0 SONT MIS A 1
7D4A 47        00510        LD      B,A       ;SAUVEGARDE CARACTERE INVERSE
7D4B 1A        00520        LD      A,(DE)    ;LECTURE CARACTERE PRECEDENT
7D4C A0        00530        AND     B         ;MIXER LES 2 CODES GRAPHIQUES
7D4D 12        00540        LD      (DE),A    ;AFFICHER LE NOUVEAU CARACTERE
7D4E 1807      00550        JR      ICI       ;PASSER AU CARACTERE SUIVANT
               00560 ;TEST
7D50 47        00570 TEST   LD      B,A       ;SAUVEGARDE CARACTERE DANS B
7D51 1A        00580        LD      A,(DE)    ;LECTURE CARACTERE PRECEDENT
7D52 A0        00590        AND     B         ;TESTER BIT PAR BIT
7D53 E63F      00600        AND     3FH       ;EFFACER LES BITS 6 ET 7
7D55 200C      00610        JR      NZ,FIN    ;Z=0 SI COLLISION
7D57 13        00620 ICI    INC     DE        ;INCREMENTER POINTEUR SUR ECRAN
7D58 18D7      00630        JR      LOOP      ;PASSER AU CARACTERE SUIVANT
7D5A E3        00640 SAUT   EX      (SP),HL   ;SAUVEGARDER HL EN PILE ET METTRE
               00650                          ;L'ADR. DE DEBUT DE LIGNE DANS HL
7D5B 114000    00660        LD      DE,64     ;NOMBRE DE CARACTERES PAR LIGNE
7D5E 19        00670        ADD     HL,DE     ;HL POINTE VERS LA LIGNE SUIVANTE
7D5F EB        00680        EX      DE,HL     ;DE POINTE VERS LA LIGNE SUIVANTE
7D60 E1        00690        POP     HL        ;RESTAURER POINTEUR SUR CHAINE
7D61 18CD      00700        JR      AFBIT     ;PASSER AU CARACTERE SUIVANT
7D63 D1        00710 FIN    POP     DE        ;DEPILER DE
7D64 C9        00720        RET               ;RETOUR
               00730 ;CHAINE GRAPHIQUE
7D65 3C        00740 DESSIN DEFB    8CH
7D66 B3        00750        DEFB    0B3H
7D67 B3        00760        DEFB    0B3H
7D68 B3        00770        DEFB    0B3H
7D69 3C        00780        DEFB    8CH
7D6A 00        00790        DEFB    0
7D00           00800        END     DEBUT
00000 TOTAL ERRORS
```

Cette méthode est très souple d'emploi et permet de réaliser pratiquement toutes les applications souhaitées. Elle est cependant plus lente que la première, elle ne se prête pas facilement à la haute résolution par générateur de caractères et elle nécessite l'initialisation de l'écran en mode graphique (remplissage de l'écran par le caractère 80 H).

3.2.3. Affichage par segment

Cet affichage fait appel aux sous programmes SET, RESET et POINT décrits au chapitre 2. La désignation d'un segment s'effectue en mettant son abscisse comprise entre 0 et 127 dans le registre E et son ordonnée comprise entre 0 et 47 dans le registre D. Cette méthode est très lente et elle s'applique difficilement aux personnages possédant un nombre important de segments. Pratiquement elle ne sera utilisée que pour le déplacement des balles contenant un ou deux segments ou pour le tracé des droites.

3.3. DÉPLACEMENT DES GRAPHISMES

Pour effectuer le déplacement d'un graphisme sur l'écran il faut, à chaque passage dans la boucle principale du jeu, effacer partiellement ou totalement le personnage de sa position initiale et le redessiner dans une position voisine. En choisissant un intervalle de déplacement suffisamment faible, l'œil du joueur ne s'apercevra pas de la nature discontinue du mouvement. En pratique un intervalle de taille égale à la largeur d'un caractère graphique donne de très bons résultats.

3.3.1. Déplacement horizontal

Sur le TRS 80 l'axe horizontal est celui pour lequel les mouvements sont les plus aisés puisque l'intervalle de déplacement est alors égal à un caractère. Pour effectuer un déplacement sur cet axe, on utilisera une variable de 16 bits contenant l'adresse du premier caractère du personnage. Cette variable sera incrémentée pour un déplacement vers la droite et décrémentée pour un déplacement vers la gauche, de façon à repérer la position du mobile. Lors d'un mouvement élémentaire l'effacement total du dessin précédent n'est pas nécessaire puisque le nouveau dessin viendra se placer en partie sur l'ancien. Seul le dernier caractère dans le sens du déplacement devra être effacé. On effectuera donc l'effacement du caractère le plus à gauche pour un déplacement vers la droite et l'effacement du caractère le plus à droite pour un déplacement vers la gauche. Ceci pourra être réalisé simplement en complétant la chaine graphique représentant le personnage par des espaces placés aux extrémités de chaque ligne. Le problème de l'effacement est ainsi résolu au prix d'une petite limitation de l'excursion du mobile.

Voici un programme permettant de déplacer horizontalement un petit vaisseau sur l'écran à l'aide des touches ← et →. Il utilise la méthode d'affichage par caractère qui est la plus appropriée étant donné la simplicité du mouvement. D'autre part la position du vaisseau est sauvegardée dans le registre double DE.

```
7A00            00010           ORG     7A00H
7A00 CDC901     00020 DEBUT     CALL    1C9H        ;EFFACEMENT DE L'ECRAN
7A03 11DE3D     00030           LD      DE,3DDEH    ;ADRESSE INITIALE DU VAISSEAU
7A06 3A4038     00040 LOOP      LD      A,(3840H);LECTURE DU CLAVIER
7A09 CB6F       00050           BIT     5,A         ;A-T-ON TAPE LA FLECHE GAUCHE ?
7A0B 200D       00060           JR      NZ,GAU      ;OUI DEPLACEMENT A GAUCHE
7A0D CB77       00070           BIT     6,A         ;A-T-ON TAPE LA FLECHE DROITE ?
7A0F 2810       00080           JR      Z,AFI       ;NON PAS DE DEPLACEMENT A DROITE
7A11 21FA3F     00090 DRT       LD      HL,3FFAH    ;FIN DE L'ECRAN -5
7A14 DF         00100           RST     18H         ;VAISSEAU EN FIN D'ECRAN ?
7A15 280A       00110           JR      Z,AFI       ;OUI IL EST DONC BLOQUE
7A17 13         00120           INC     DE          ;DEPLACEMENT VERS LA DROITE
7A18 1807       00130           JR      AFI         ;AFFICHAGE DU VAISSEAU
7A1A 21003C     00140 GAU       LD      HL,3C00H    ;DEBUT DE L'ECRAN
7A1D DF         00150           RST     18H         ;VAISSEAU EN DEBUT D'ECRAN ?
7A1E 2801       00160           JR      Z,AFI       ;OUI IL EST DONC BLOQUE
7A20 1B         00170           DEC     DE          ;DEPLACEMENT VERS LA GAUCHE
7A21 21397A     00180 AFI       LD      HL,DESSIN;ADRESSE DE LA CHAINE GRAPHIQUE
7A24 D5         00190           PUSH    DE          ;SAUVEGARDER POSITION VAISSEAU
7A25 CD317A     00200           CALL    MES         ;AFFICHAGE VAISSEAU
7A28 D1         00210           POP     DE          ;RESTAURER POSITION VAISSEAU
7A29 010005     00220           LD      BC,500H     ;COMPTEUR BOUCLE D'ATTENTE
7A2C CD6000     00230           CALL    60H         ;ATTENTE
7A2F 18D5       00240           JR      LOOP        ;BOUCLER
7A31 7E         00250 MES       LD      A,(HL)      ;SOUS PROGRAMME D'AFFICHAGE: MES
7A32 B7         00260           OR      A           ;
7A33 C8         00270           RET     Z           ;
7A34 12         00280           LD      (DE),A      ;
7A35 13         00290           INC     DE          ;
7A36 23         00300           INC     HL          ;
7A37 18F8       00310           JR      MES         ;
                00320 ;CHAINE GRAPHIQUE
7A39 80         00330 DESSIN    DEFB    80H         ;ESPACE
7A3A 9D         00340           DEFB    9DH         ;
7A3B B7         00350           DEFB    0B7H        ;
7A3C BB         00360           DEFB    0BBH        ;
7A3D AE         00370           DEFB    0AEH        ;
7A3E 80         00380           DEFB    80H         ;ESPACE
7A3F 00         00390           DEFB    0           ;ZERO POUR FIN DE CHAINE
7A00            00400           END     DEBUT
00000 TOTAL ERRORS
```

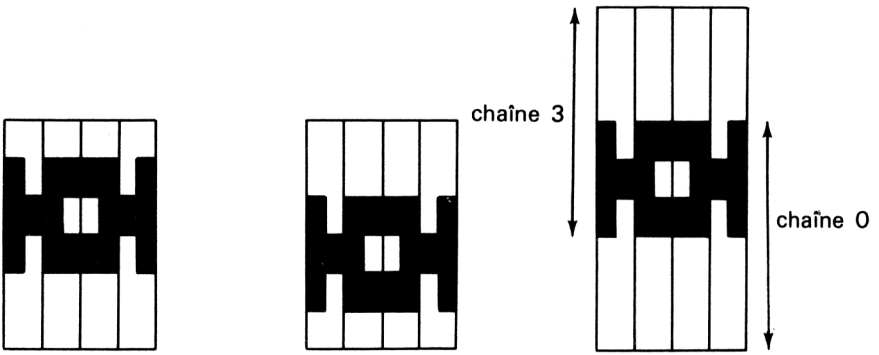
3.3.2. Déplacement vertical

Sur le TRS 80 le déplacement suivant l'axe vertical est plus difficile à réaliser puisque les caractères graphiques ont une hauteur trois fois plus grande que leur largeur. Aussi pour conserver la continuité du mouvement il faudra se déplacer d'un tiers de caractère à chaque fois. Le passage d'une ligne de l'écran à une autre s'effectuera donc au bout de trois déplacements élémentaires.

Pour réaliser ce mouvement, le plus simple sera de disposer de trois chaînes graphiques représentant le même personnage décalé d'un tiers de caractère. Au cours du déplacement, on affichera successivement et dans le bon ordre les trois chaînes, puis on sautera de ligne et on recommencera avec la première chaîne. Il sera utile pour celà de disposer d'un

compteur indiquant le numéro de chaîne graphique courante afin de savoir quand doit être effectué le saut de ligne.

La façon la plus simple d'assurer l'effacement du personnage précédent lors d'un déplacement élémentaire consiste à le faire disparaître totalement de l'écran en appelant le sous programme d'effacement contenu dans AFBIT. Cependant il peut être intéressant d'étudier une façon plus directe et donc plus rapide d'y arriver. Celle-ci se bornera à effacer les derniers caractères du personnage quand cela sera vraiment nécessaire. L'effacement sera assuré par l'adjonction d'espaces de part et d'autre d'une des chaînes.



Pour un déplacement vers le bas, on dessinera successivement la chaîne 1, la chaîne 2, la chaîne 3, on sautera à la ligne du dessous et on recommencera par la chaîne 1. Pour un déplacement vers le haut on dessinera la chaîne 2, la chaîne 1 et la chaîne 0 avant de passer à la ligne du dessus et de recommencer par la chaîne 2. Les espaces situés dans les chaînes 0 et 3 assureront l'effacement du personnage précédent.

Le programme donné ci-dessous utilise cette méthode pour déplacer un petit vaisseau grâce aux flèches ↑ et ↓. Les adresses des quatre chaînes sont placées dans un tableau parcouru par un pointeur contenu dans le registre IX, afin de simplifier la procédure d'affichage. Le registre DE est utilisé pour stocker la position du vaisseau, c'est-à-dire l'adresse du caractère le plus en haut et à gauche. D'autre part le programme effectue les tests nécessaires pour conserver le vaisseau sur l'écran.

```
7F00            00010           ORG     7F00H
7F00 CDC901     00020 DEBUT     CALL    1C9H        ;EFFACEMENT DE L'ECRAN
7F03 11DE3D     00030           LD      DE,3DDEH    ;ADRESSE INITIALE DU VAISSEAU
7F06 DD21827F   00040           LD      IX,NUM1     ;INITIALISATION DU POINTEUR
7F0A 1850       00050           JR      AFI         ;AFFICHAGE INITIAL
7F0C 3A4038     00060 LOOP      LD      A,(3840H);LECTURE CLAVIER
7F0F CB67       00070           BIT     4,A         ;A-T-ON TAPE SUR LA FLECHE BAS ?
7F11 2029       00080           JR      NZ,BAS      ;OUI DEPLACEMENT VERS LE BAS
7F13 CB5F       00090           BIT     3,A         ;A-T-ON TAPE SUR LA FLECHE HAUT ?
7F15 284E       00100           JR      Z,WAIT      ;NON PAS DE DEPLACEMENT EN HAUT
7F17 DDE5       00110 HAUT      PUSH    IX          ;LE POINTEUR SUR LE TABLEAU DES
7F19 E1         00120           POP     HL          ;CHAINES EST MIS DANS HL
7F1A D5         00130           PUSH    DE          ;SAUVEGARDER POSITION VAISSEAU
7F1B 11807F     00140           LD      DE,NUM0     ;ADRESSE DU DEBUT DU TABLEAU
7F1E DF         00150           RST     18H         ;LE POINTEUR EST-IL AU DEBUT ?
7F1F D1         00160           POP     DE          ;RESTAURER POSITION VAISSEAU
7F20 2806       00170           JR      Z,A0        ;SI OUI REINITIALISER POINTEUR
7F22 DD2B       00180           DEC     IX          ;DECREMENTER LE POINTEUR DE 2
7F24 DD2B       00190           DEC     IX          ;POUR CHAINE SUIVANTE
7F26 1834       00200           JR      AFI         ;AFFICHAGE
7F28 213F3C     00210 A0        LD      HL,3C3FH    ;FIN DE LA PREMIERE LIGNE
7F2B DF         00220           RST     18H         ;VAISSEAU DANS LA PREMIERE LIGNE ?
7F2C 3037       00230           JR      NC,WAIT     ;SI OUI IL EST BLOQUE
7F2E DD21847F   00240           LD      IX,NUM2     ;REINITIALISER LE POINTEUR
7F32 EB         00250           EX      DE,HL       ;POSITION DU VAISSEAU DANS HL
7F33 114000     00260           LD      DE,64       ;64 CARACTERES PAR LIGNE
7F36 B7         00270           OR      A           ;CARRY=0
7F37 ED52       00280           SBC     HL,DE       ;PASSER A LA LIGNE PRECEDENTE
7F39 EB         00290           EX      DE,HL       ;POSITION DU VAISSEAU DANS DE
7F3A 1820       00300           JR      AFI         ;AFFICHAGE
7F3C DDE5       00310 BAS       PUSH    IX          ;LE POINTEUR SUR LE TABLEAU DES
7F3E E1         00320           POP     HL          ;CHAINES EST MIS DANS HL
7F3F D5         00330           PUSH    DE          ;SAUVEGARDER POSITION VAISSEAU
7F40 11867F     00340           LD      DE,NUM3     ;ADRESSE DE FIN DU TABLEAU
7F43 DF         00350           RST     18H         ;LE POINTEUR EST-IL A LA FIN ?
7F44 D1         00360           POP     DE          ;RESTAURER POSITION
7F45 2806       00370           JR      Z,A1        ;SI OUI REINITIALISER LE POINTEUR
7F47 DD23       00380           INC     IX          ;AUGMENTER LE POINTEUR DE 2 POUR
7F49 DD23       00390           INC     IX          ;LA CHAINE SUIVANTE
7F4B 180F       00400           JR      AFI         ;AFFICHAGE
7F4D 217F3F     00410 A1        LD      HL,3F7FH    ;FIN DE L'AVANT-DERNIERE LIGNE
7F50 DF         00420           RST     18H         ;VAISSEAU DANS CETTE LIGNE ?
7F51 3812       00430           JR      C,WAIT      ;SI OUI IL EST BLOQUE
7F53 DD21827F   00440           LD      IX,NUM1     ;REINITIALISER LE POINTEUR
7F57 214000     00450           LD      HL,64       ;64 CARACTERES PAR LIGNE
7F5A 19         00460           ADD     HL,DE       ;PASSER A LA LIGNE SUIVANTE
7F5B EB         00470           EX      DE,HL       ;POSITION DANS DE
7F5C DD6E00     00480 AFI       LD      L,(IX+0)    ;POIDS FAIBLE ADRESSE CHAINE
7F5F DD6601     00490           LD      H,(IX+1)    ;POIDS FORT
7F62 CD6D7F     00500           CALL    AFF         ;AFFICHAGE
7F65 010010     00510 WAIT      LD      BC,1000H    ;COMPTEUR DE BOUCLE
7F68 CD6000     00520           CALL    60H         ;ATTENTE
7F6B 189F       00530           JR      LOOP        ;BOUCLER
7F6D D5         00540 AFF       PUSH    DE          ;SAUVEGARDER POSITION
7F6E 010400     00550           LD      BC,4        ;4 CARACTERES A AFFICHER
7F71 EDB0       00560           LDIR                ;AFFICHAGE
7F73 EB         00570           EX      DE,HL       ;POSITION DANS HL
7F74 013C00     00580           LD      BC,60       ;PASSER A LA LIGNE DU DESSOUS
7F77 09         00590           ADD     HL,BC       ;
7F78 EB         00600           EX      DE,HL       ;POSITION DANS DE
7F79 010400     00610           LD      BC,4        ;4 CARACTERES A AFFICHER
7F7C EDB0       00620           LDIR                ;AFFICHAGE
7F7E D1         00630           POP     DE          ;RESTAURER POSITION
7F7F C9         00640           RET                 ;
                00650 ;
```

```
                00660 ;TABLEAU DES ADRESSES DES CHAINES GRAPHIQUES
                00670 ;
7F80 8C7F       00680 NUM0      DEFW    CH0         ;ADRESSE CHAINE 0
7F82 947F       00690 NUM1      DEFW    CH1         ;ADRESSE CHAINE 1
7F84 9C7F       00700 NUM2      DEFW    CH2         ;ADRESSE CHAINE 2
7F86 887F       00710 NUM3      DEFW    CH3         ;ADRESSE CHAINE 3
                00720 ;
                00730 ;CHAINES GRAPHIQUES
                00740 ;
7F88 8080       00750 CH3       DEFW    8080H
7F8A 8080       00760           DEFW    8080H
7F8C 9DB7       00770 CH0       DEFW    0B79DH
7F8E BBAE       00780           DEFW    0AEBBH
7F90 8080       00790           DEFW    8080H
7F92 8080       00800           DEFW    8080H
7F94 B49C       00810 CH1       DEFW    9CB4H
7F96 ACB8       00820           DEFW    0B8ACH
7F98 8183       00830           DEFW    8381H
7F9A 8382       00840           DEFW    8283H
7F9C 90B0       00850 CH2       DEFW    0B090H
7F9E B0A0       00860           DEFW    0A0B0H
7FA0 878D       00870           DEFW    8D87H
7FA2 8E8B       00880           DEFW    8B8EH
7F00            00890           END     DEBUT
00000 TOTAL ERRORS
```

Les trois chaînes graphiques pourront être choisies légèrement différentes pour simuler un mouvement du personnage pendant son déplacement.

3.3.3. Déplacement dans une direction quelconque

Pour effectuer un mouvement dans une des quatre diagonales de l'écran, il faudra combiner l'effet du déplacement vertical et du déplacement horizontal. Par exemple, pour se déplacer sur la diagonale partant du bas à gauche et allant en haut à droite, on fera en même temps un mouvement vers le haut et un mouvement vers la droite. L'affichage et l'effacement des graphismes seront réalisés de préférence par le sous programme AFBIT qui est le mieux adapté à un déplacement quelconque.

Le programme ci-dessous permet de déplacer notre vaisseau spatial dans les huit directions principales de l'écran grâce aux touches ←, →, ↑ et ↓ . Le déplacement en diagonale sera obtenu par l'appui sur deux flèches de directions distinctes.

```
7E00            00010           ORG     7E00H
7E00 CDC77E     00020 DEBUT     CALL    CLS         ;EFFACEMENT DE L'ECRAN
7E03 11DE3D     00030           LD      DE,3DDEH    ;ADRESSE INITIALE DU VAISSEAU
7E06 DD21D77E   00040           LD      IX,NUM1     ;INITIALISATION DU POINTEUR
7E0A 187D       00050           JR      AFI         ;AFFICHAGE INITIAL
7E0C 3A4038     00060 LOOP      LD      A,(3840H);LECTURE CLAVIER
7E0F E678       00070           AND     78H         ;A-T-ON TAPE UNE DES 4 FLECHES ?
7E11 CA947E     00080           JP      Z,WAIT      ;NON, PAS DE DEPLACEMENT
7E14 F5         00090           PUSH    AF          ;SAUVEGARDER A
7E15 DD6E00     00100           LD      L,(IX+0)    ;POIDS FAIBLE ADRESSE CHAINE
7E18 DD6601     00110           LD      H,(IX+1)    ;POIDS FORT
7E1B 0E01       00120           LD      C,1         ;CODE EFFACEMENT
7E1D CD9D7E     00130           CALL    AFBIT       ;EFFACEMENT
7E20 F1         00140           POP     AF          ;RESTAURER A
7E21 CB67       00150           BIT     4,A         ;A-T-ON TAPE SUR LA FLECHE BAS ?
7E23 2029       00160           JR      NZ,BAS      ;OUI DEPLACEMENT VERS LE BAS
7E25 CB5F       00170           BIT     3,A         ;A-T-ON TAPE SUR LA FLECHE HAUT ?
7E27 2845       00180           JR      Z,B0        ;NON PAS DE DEPLACEMENT EN HAUT
7E29 DDE5       00190 HAUT      PUSH    IX          ;LE POINTEUR SUR LE TABLEAU DES
7E2B E1         00200           POP     HL          ;CHAINES EST MIS DANS HL
7E2C D5         00210           PUSH    DE          ;SAUVEGARDER POSITION VAISSEAU
7E2D 11D57E     00220           LD      DE,NUM0     ;ADRESSE DU DEBUT DU TABLEAU
7E30 DF         00230           RST     18H         ;LE POINTEUR EST-IL AU DEBUT ?
7E31 D1         00240           POP     DE          ;RESTAURER POSITION VAISSEAU
7E32 2806       00250           JR      Z,A0        ;SI OUI REINITIALISER POINTEUR
7E34 DD2B       00260           DEC     IX          ;DECREMENTER LE POINTEUR DE 2
7E36 DD2B       00270           DEC     IX          ;POUR CHAINE SUIVANTE
7E38 1834       00280           JR      B0          ;SUITE
7E3A 213F3C     00290 A0        LD      HL,3C3FH    ;FIN DE LA PREMIERE LIGNE
7E3D DF         00300           RST     18H         ;VAISSEAU DANS LA PREMIERE LIGNE ?
7E3E 302E       00310           JR      NC,B0       ;SI OUI IL EST BLOQUE
7E40 DD21D97E   00320           LD      IX,NUM2     ;REINITIALISER LE POINTEUR
7E44 EB         00330           EX      DE,HL       ;POSITION DU VAISSEAU DANS HL
7E45 114000     00340           LD      DE,64       ;64 CARACTERES PAR LIGNE
7E48 B7         00350           OR      A           ;CARRY=0
7E49 ED52       00360           SBC     HL,DE       ;PASSER A LA LIGNE PRECEDENTE
7E4B EB         00370           EX      DE,HL       ;POSITION DU VAISSEAU DANS DE
7E4C 1820       00380           JR      B0          ;SUITE
7E4E DDE5       00390 BAS       PUSH    IX          ;LE POINTEUR SUR LE TABLEAU DES
7E50 E1         00400           POP     HL          ;CHAINES EST MIS DANS HL
7E51 D5         00410           PUSH    DE          ;SAUVEGARDER POSITION VAISSEAU
7E52 11D97E     00420           LD      DE,NUM2     ;ADRESSE DE FIN DU TABLEAU
7E55 DF         00430           RST     18H         ;LE POINTEUR EST-IL A LA FIN ?
7E56 D1         00440           POP     DE          ;RESTAURER POSITION
7E57 2806       00450           JR      Z,A1        ;SI OUI REINITIALISER LE POINTEUR
7E59 DD23       00460           INC     IX          ;AUGMENTER LE POINTEUR DE 2 POUR
7E5B DD23       00470           INC     IX          ;LA CHAINE SUIVANTE
7E5D 180F       00480           JR      B0          ;SUITE
7E5F 217F3F     00490 A1        LD      HL,3F7FH    ;FIN DE L'AVANT-DERNIERE LIGNE
7E62 DF         00500           RST     18H         ;VAISSEAU DANS CETTE LIGNE ?
7E63 3809       00510           JR      C,B0        ;SI OUI IL EST BLOQUE
7E65 DD21D57E   00520           LD      IX,NUM0     ;REINITIALISER LE POINTEUR
7E69 214000     00530           LD      HL,64       ;64 CARACTERES PAR LIGNE
7E6C 19         00540           ADD     HL,DE       ;PASSER A LA LIGNE SUIVANTE
7E6D EB         00550           EX      DE,HL       ;POSITION DANS DE
7E6E 3A4038     00560 B0        LD      A,(3840H);LECTURE CLAVIER
7E71 CB6F       00570           BIT     5,A         ;A-T-ON TAPE LA FLECHE GAUCHE ?
7E73 200E       00580           JR      NZ,GAU      ;OUI DEPLACEMENT A GAUCHE
7E75 CB77       00590           BIT     6,A         ;A-T-ON TAPE LA FLECHE DROITE ?
7E77 2810       00600           JR      Z,AFI       ;NON PAS DE DEPLACEMENT A DROITE
7E79 7B         00610 DRT       LD      A,E         ;POIDS FAIBLE POSITION VAISSEAU
7E7A E63F       00620           AND     3FH         ;POSITION DANS LA LIGNE
7E7C FE3C       00630           CP      3CH         ;EST-ON EN BOUT DE LIGNE ?
7E7E 2809       00640           JR      Z,AFI       ;OUI ON EST BLOQUE
7E80 13         00650           INC     DE          ;AVANCER D'UN CARACTERE
7E81 1806       00660           JR      AFI         ;AFFICHAGE
7E83 7B         00670 GAU       LD      A,E         ;POIDS FAIBLE POSITION VAISSEAU
7E84 E63F       00680           AND     3FH         ;EST-ON EN DEBUT DE LIGNE ?
7E86 2801       00690           JR      Z,AFI       ;OUI ON EST BLOQUE
7E88 1B         00700           DEC     DE          ;RECULER D'UN CARACTERE
7E89 DD6E00     00710 AFI       LD      L,(IX+0)    ;POIDS FAIBLE ADRESSE CHAINE
```

```
7E8C DD6601    00720          LD      H,(IX+1) ;POIDS FORT
7E8F 0E00      00730          LD      C,0      ;CODE AFFICHAGE
7E91 CD9D7E    00740          CALL    AFBIT    ;AFFICHAGE
7E94 010010    00750 WAIT     LD      BC,1000H ;COMPTEUR DE BOUCLE
7E97 CD6000    00760          CALL    60H      ;ATTENTE
7E9A C30C7E    00770          JP      LOOP     ;BOUCLER
               00780 ;
               00790 ;SOUS PROGRAMME D'AFFICHAGE PAR BIT
               00800 ;
7E9D D5        00810 AFBIT    PUSH    DE       ;SAUVEGARDER POSITION
7E9E D5        00820 C0       PUSH    DE       ;
7E9F 7E        00830 C1       LD      A,(HL)   ;
7EA0 23        00840          INC     HL       ;
7EA1 B7        00850          OR      A        ;
7EA2 2820      00860          JR      Z,FIN    ;
7EA4 FE01      00870          CP      1        ;
7EA6 2813      00880          JR      Z,SAUT   ;
7EA8 CB41      00890          BIT     0,C      ;
7EAA 2006      00900          JR      NZ,EFFA  ;
7EAC 47        00910 AFFI     LD      B,A      ;
7EAD 1A        00920          LD      A,(DE)   ;
7EAE B0        00930          OR      B        ;
7EAF 12        00940          LD      (DE),A   ;
7EB0 1806      00950          JR      ICI      ;
7EB2 EE3F      00960 EFFA     XOR     3FH      ;
7EB4 47        00970          LD      B,A      ;
7EB5 1A        00980          LD      A,(DE)   ;
7EB6 A0        00990          AND     B        ;
7EB7 12        01000          LD      (DE),A   ;
7EB8 13        01010 ICI      INC     DE       ;
7EB9 18E4      01020          JR      C1       ;
7EBB E3        01030 SAUT     EX      (SP),HL  ;
7EBC 114000    01040          LD      DE,64    ;
7EBF 19        01050          ADD     HL,DE    ;
7EC0 EB        01060          EX      DE,HL    ;
7EC1 E1        01070          POP     HL       ;
7EC2 18DA      01080          JR      C0       ;
7EC4 D1        01090 FIN      POP     DE       ;
7EC5 D1        01100          POP     DE       ;RESTAURER POSITION
7EC6 C9        01110          RET              ;
7EC7 21003C    01120 CLS      LD      HL,3C00H ;EFFACEMENT DE L'ECRAN
7ECA 11013C    01130          LD      DE,3C01H ;
7ECD 3680      01140          LD      (HL),80H ;
7ECF 01FF03    01150          LD      BC,3FFH  ;
7ED2 EDB0      01160          LDIR             ;
7ED4 C9        01170          RET              ;
               01180 ;
               01190 ;TABLEAU DES ADRESSES DES CHAINES GRAPHIQUES
               01200 ;
7ED5 DB7E      01210 NUM0     DEFW    CH0      ;ADRESSE CHAINE 0
7ED7 E07E      01220 NUM1     DEFW    CH1      ;ADRESSE CHAINE 1
7ED9 EA7E      01230 NUM2     DEFW    CH2      ;ADRESSE CHAINE 2
               01240 ;
               01250 ;CHAINES GRAPHIQUES
               01260 ;
7EDB 9DB7      01270 CH0      DEFW    0B79DH
7EDD BBAE      01280          DEFW    0AEBBH
7EDF 00        01290          DEFB    0
7EE0 B49C      01300 CH1      DEFW    9CB4H
7EE2 ACB8      01310          DEFW    0B8ACH
7EE4 01        01320          DEFB    1
7EE5 8183      01330          DEFW    8381H
7EE7 8382      01340          DEFW    8283H
7EE9 00        01350          DEFB    0
7EEA 90B0      01360 CH2      DEFW    0B090H
7EEC B0A0      01370          DEFW    0A0B0H
7EEE 01        01380          DEFB    1
7EEF 878D      01390          DEFW    8D87H
7EF1 8E8B      01400          DEFW    8B8EH
7EF3 00        01410          DEFB    0
7E00           01420          END     DEBUT
00000 TOTAL ERRORS
```

Un déplacement dans une direction quelconque sera constitué d'une suite de mouvements élémentaires dont la direction est l'une des huit directions principales décrites précédemment. La détermination des mouvements élémentaires à effectuer se fera en utilisant un algorithme analogue à celui employé pour tracer une droite sur l'écran. Cet algorithme permet, à partir d'un point de la droite, de déterminer le point suivant de telle sorte qu'il soit aligné avec le reste de la droite. La distance entre ces deux points est au plus égale à un segment graphique dans le sens vertical et dans le sens horizontal. Par conséquent le chemin entre ces deux points donne le mouvement élémentaire à effectuer pour se déplacer sur la droite. Cependant il faudra faire attention au fait que cet algorithme utilisait dans la direction horizontale la largeur d'un segment graphique et non pas celle d'un caractère. Étant donné que l'unité de base du déplacement horizontal est le caractère, il faudra employer comme abscisse non pas le numéro du segment dans la ligne, mais le numéro du caractère ($0 \leqslant X_1 \leqslant 63$, $0 \leqslant X_2 \leqslant 63$).

Dans la plupart des jeux d'animation, les mouvements se feront dans une des huit directions principales ; aussi nous ne nous attarderons pas plus sur les déplacements dans une direction quelconque qui ne devraient pas poser de difficultés de réalisation.

3.3.4. Déplacement par effacement et réécriture systématique de l'écran

Pour simplifier la procédure d'effacement des personnages précédents et pour faciliter les tests de déplacement, certains programmes ont adopté un procédé radical mettant à profit la rapidité du microprocesseur. Au début de la boucle principale du jeu, ils effacent totalement la partie de l'écran où se produit les mouvements. Ensuite, ils redessinent tous les personnages, décalés ou non de leur position d'origine. Les programmes de déplacement sont semblables mis à part le fait qu'ils ne contiennent pas les phases délicates d'effacement.

Cette méthode est d'un emploi plus aisé que les précédentes et elle s'adapte facilement aux déplacements complexes. Elle permet en plus de faire se mouvoir plusieurs personnages les uns au-dessus des autres pour simuler un effet de profondeur. Cependant elle est beaucoup plus lente, ce qui a pour effet de diminuer le temps accordé à la production du son.

D'autre part, elle augmente le nombre d'écritures sur la mémoire écran ce qui accroît le phénomène de traits noirs apparaissant sur les TRS 80 modèle I et diminue la stabilité de l'image. Enfin elle nécessite l'emploi d'instructions supplémentaires pour assurer l'affichage des graphismes fixes (décors, personnages immobiles). Cette méthode s'applique assez bien sur un TRS 80 qui possède un graphisme peu évolué et donc rapidement modifiable. Sur d'autres ordinateurs fonctionnant en haute résolution, il y aura intérêt à diminuer les affichages inutiles et cette méthode sera à proscrire.

3.4. CHOIX DES VARIABLES ET DES TABLEAUX NÉCESSAIRES POUR CARACTÉRISER LES PERSONNAGES

Dans les procédés que nous avons décrits précédemment, nous avons implicitement utilisé une variable de seize bits contenue dans le registre DE pour représenter la position du vaisseau. Dans un programme de jeu complet, il faudra pouvoir disposer de la position et de différents paramètres propres à chaque mobile. Aussi il faudra commencer par déterminer quelles sont les informations nécessaires à caractériser chaque type de personnage. Ensuite il faudra choisir le nombre d'octets à utiliser pour stocker ces informations. Ce nombre d'octets pourra varier de façon importante selon la complexité du jeu et l'habileté du programmeur. Un envahisseur pourra par exemple nécessiter l'emploi de trois octets : deux pour la position et un pour sa valeur en points. Les variables correspondantes à des mobiles de même type seront regroupées dans un tableau pour être exploitées séquentiellement par le sous programme de déplacement. Généralement le nombre de cases dans le tableau sera supérieur au nombre de mobiles sur l'écran. Aussi il faudra prévoir un moyen pour savoir si une case correspond réellement à un personnage sur l'écran. On pourra, par exemple, mettre zéro comme position si la case ne fait référence à aucun des personnages. Toutes les variables et les tableaux seront stockés à des adresses fixes en mémoire grâce à la directive DEFS.

3.5. TESTS DE DÉPLACEMENT

La première catégorie de tests concerne les limites physiques de l'écran. Lors du déplacement il faut s'assurer que les mobiles restent dans les limites de l'écran afin de ne pas empiéter sur les mémoires vives qui se trouvent juste après. Ce test est absolument nécessaire pour le vaisseau du joueur qui ne pourra se déplacer qu'entre deux bornes. Pour les autres personnages on pourra soit limiter leur excursion de façon à ce qu'il n'y ait jamais débordement, soit les faire disparaître du jeu au moment où ils sortent de l'écran. Certains programmes utilisent même un algorithme qui donne l'impression que l'écran est sphérique en faisant réapparaître sur le côté opposé les objets qui sortent d'un côté de l'écran.

La deuxième catégorie de tests assure la gestion des rencontres entre les différents mobiles. Ces tests pourront se faire de différentes manières suivant le procédé d'affichage utilisé. Ils pourront s'effectuer à chaque déplacement par l'examen des positions respectives des divers mobiles données par leurs variables associées. Cette méthode est assez lente si le nombre de personnages est important, mais c'est la seule qui puisse être utilisée avec le procédé d'affichage effectuant des effacements et des réécritures totales de l'écran. Si l'affichage se fait par caractères, on pourra à chaque déplacement examiner le contenu des mémoires où le mobile va se placer. Si elles ne contiennent pas que des espaces, il faudra alors déterminer d'où viennent les autres caractères afin d'agir en conséquence. Si on utilise AFBIT pour l'affichage, il faudra appeler le sous programme de test avant de dessiner le personnage. Si le test indique la présence d'un corps étranger, il y aura lieu de parcourir toutes les variables de position pour déterminer sa provenance et faire exploser le personnage si nécessaire.

Toutes ces opérations de test seront de préférence intégrées au sous programme de déplacement pour conserver la structure du jeu explicitée dans le chapitre 1. Ces tests constituent en général la partie la plus délicate du programme. Il arrive fréquemment que des tests soient mal faits ou tout simplement oubliés, ce qui conduit à un disfonctionnement dans certains cas de figure. Ces phénomènes arrivent de façon aléatoire, il est difficile de déceler d'où ils proviennent et donc de les corriger. Aussi il y aura intérêt à s'assurer que toutes les situations possibles ont bien été envisagées lors de l'analyse du jeu.

3.6. CONSEILS POUR ABORDER UN JEU PARTICULIER

Lorsqu'on aborde un programme de jeu, il est utile de commencer par dessiner sur une feuille de papier les graphismes qui seront affichés sur l'écran. Pour cela vous pourrez utiliser une feuille spéciale, quadrillée selon les divisions de l'écran. Au vu de cette feuille, vous déterminerez les mouvements des différents personnages et les tests à effectuer. Vous aurez ensuite intérêt à simplifier les déplacements au prix de quelques limitations du jeu pour que le programme ne soit pas trop dur à réaliser. Parallèlement à ces déplacements, il faudra étudier les sous programmes permettant de faire naître de nouveaux mobiles (tir, apparition aléatoire d'un envahisseur). Une fois cela effectué, vous pourrez vous occuper des autres sous programmes moins indispensables (explosion des vaisseaux, compte des points) qui pourrait être simplifiés dans un premier temps.

Le programme de jeu SONAR que nous avons choisi comme exemple dans ce livre est d'une complexité moyenne afin de ne pas décourager les lecteurs tout en montrant les différents problèmes qui se posent au programmeur. Si vous êtes plutôt débutant en programmation, cet exemple peut vous paraître trop difficile. Vous aurez alors intérêt à vous inititier par des jeux plus simples mettant en prise un faible nombre de mobiles (mur de brique, flipper, ping-pong, etc...)

3.7. APPLICATION A NOTRE EXEMPLE

Le listing commenté du programme SONAR est donné intégralement en annexe. Aussi nous nous contenterons, dans cette partie, d'expliquer le fonctionnement des différents sous programmes de déplacement en indiquant la structure des variables utilisées pour chaque mobile. Il sera bon à ce sujet d'examiner le dessin des différents personnages et leur implantation sur l'écran en regardant le schéma de la page suivante qui représente l'état de l'écran au cours du jeu.

SCORE = 036400 MEILLEUR SCORE = 000000 BATEAUX RESTANTS: 3
Sens
400
100

3.7.1. Déplacement du bateau

Le bateau piloté par le joueur occupe deux lignes entières. Aussi il est représenté par une chaîne contenant un caractère 1 indiquant un saut de ligne. L'affichage du bateau est assuré par le sous programme DBA qui commence par dessiner la ligne du dessous puis la ligne du dessus. Le sous programme ABA effectue le déplacement du bateau vers la droite quand le joueur appuie sur la touche →, et vers la gauche quand il appuie sur ←. La variable PBA est utilisée pour sauvegarder la position du vaisseau c'est-à-dire l'adresse la plus à gauche de la seconde ligne du dessin du bateau. L'effacement du navire précédent est assuré par l'ajout d'espaces et de caractères de code 5FH (surface de la mer) de part et d'autre de la chaîne graphique. Lors du déplacement on vérifie que le bateau ne dépasse pas les limites de l'écran et qu'il ne vient pas se placer sur une case où explose une mine. Si ce dernier cas se produit, le contrôle est passé au programme BOUM qui fait exploser le bateau en le redessinant successivement dans les différentes phases de son naufrage, à l'instar des dessins animés. Ensuite ce programme efface toutes les bombes et les mines et redessine la surface de la mer avec le bateau dessus.

La variable NBA contenant le nombre de bateaux du joueur est diminuée d'une unité. Si cette variable est non nulle la partie continue. Dans le cas contraire, le message "TERMINE" s'affiche sur l'écran et on retourne à la boucle principale après avoir positionné l'indicateur de fin de partie contenu dans FPA. Pendant un instant on conserve l'affichage du jeu et le déplacement des sous-marins, afin de laisser le temps au joueur de noter son score. La variable CNT incrémentée à chaque passage dans la boucle principale est utilisée pour limiter la durée de cet instant et repasser à la présentation. L'indicateur FPA permet d'empêcher le déplacement des autres mobiles pendant cette phase.

3.7.2. Déplacement des sous-marins

Les sous-marins peuvent se déplacer soit vers la gauche soit vers la droite. Dans chacun des cas on utilise un dessin de sous-marin particulier puisque ceux-ci ne sont pas symétriques. Pour simplifier les procédures de déplacement nous avons limité les possibilités d'évolution des

sous-marins. Les lignes 4, 6, 8, 10, 12, 14 de l'écran sont réservées aux sous-marins allant à gauche et les lignes 5, 7, 9, 11, 13, 15 à ceux qui vont à droite (voir les flèches sur le bord droit du schéma de l'écran). D'autre part, on ne peut avoir qu'un seul vaisseau par ligne. Les variables associées aux sous-marins sont stockées dans un tableau qui se trouve en parallèle avec les numéros de ligne. Le premier élément du tableau correspond à la ligne 4 et le dernier à la ligne 15. Chaque élément est constitué de quatre octets. Si le premier octet est nul, il indique l'absence du sous-marin. Dans le cas contraire il sert de compteur pour savoir quand doit être effectué le déplacement. Le second octet est utilisé pour initialiser le premier octet à chaque mouvement élémentaire. Il contient donc le rapport de vitesse entre le bateau et le sous-marin. Ce rapport permet, par une simple soustraction, de connaître la valeur en points du sous-marin, puisque ces deux quantités sont dépendantes. Les deux octets suivants indiquent la position du sous-marin, à savoir l'octet le plus à gauche sur le dessin.

octet 1	octet 2	octet 3	octet 4
∅ → Sous-marin inexistant Sinon compteur de déplacement	Rapport de vitesse bateau/sous-marin valeur en points = 1 + P1 − octet 2	Poids faible	Poids fort
		Position du sous-marin (octet à gauche du dessin)	

La génération des sous-marins se fait de façon aléatoire par le sous programme SSM qui utilise les paramètres PØ et P1 lors de l'appel de la fonction RND. La création se limite à l'initialisation des quatre octets d'un des éléments du tableau, le numéro de ligne et la vitesse étant tirés au sort.

Le sous programme ASM assure le déplacement de tous les sous-marins en distinguant ceux qui vont à droite de ceux qui vont à gauche. Le dessin de ces deux types de vaisseaux est réalisé par les sous-programmes DSD et DSG qui exploitent la structure de l'écran pour donner l'impression que les sous-marins sortent d'un bord de l'écran et rentrent dans un autre de manière continue (test du bit six de l'adresse de la mémoire écran). Lors d'un déplacement élémentaire, l'effacement du dessin précédent se fait en ajoutant un espace dans la chaîne graphique à l'arrière de chaque type de sous-marin. Des tests sont effectués pour

déterminer à quel moment un sous-marin disparaît de l'écran. Quand cela se produit, on met à zéro le premier octet de l'élément du tableau qui le caractérise afin d'indiquer que la ligne est vide. On fait, d'autre part, des tests pour s'assurer que le sous-marin ne rencontre pas un objet étranger sur son passage. S'il rencontre une mine, le sous programme ETO est appelé pour indiquer que la mine correspondante est cachée. S'il rencontre une bombe, celle-ci est détruite (sous programme EBM) et le sous-marin explose. L'explosion est assurée par les sous programmes ESD et ESG qui commencent par dessiner le vaisseau en inversé pendant un court instant, puis le font disparaître. A sa place, on affiche le nombre de points gagnés. Ce nombre est inscrit en principe au milieu du sous-marin, sauf si celui-ci était au bord de l'écran, auquel cas ces chiffres sont placés au bord de l'écran. Chaque fois que le joueur gagne des points, le sous programme CMT est appelé pour les comptabiliser. C'est ce sous programme qui accorde un bateau supplémentaire au joueur lorsqu'il atteint 20 000 points. A la fin de l'explosion, la variable caractérisant le sous-marin est initialisée comme suit: le premier octet contient dix, le bit sept du second octet est mis à 1 et les deux derniers octets contiennent l'adresse du nombre qui est affiché sur l'écran. Ainsi, pendant la durée de dix boucles principales, ce nombre restera affiché pour que le joueur ait le temps de le noter. Ensuite, le bit sept du second octet indiquera au sous-programme ASM que le sous-marin vient d'exploser. Le nombre sera alors effacé et le premier octet de la variable sera mis à zéro pour dire que la ligne est vide.

3.7.3. Déplacement des bombes

Les bombes sont constituées de deux segments côte à côte et se déplacent verticalement de haut en bas d'un segment à chaque fois. Pour simplifier les procédures de déplacement, nous avons imposé que toutes les bombes soient, à un instant donné, sur le même segment graphique dans un caractère. Aussi, pour l'affichage, nous utilisons un procédé très simple faisant appel à la variable PCA qui indique le segment graphique à dessiner dans chacune des cases contenant une bombe. Un tableau de cinq éléments est utilisé pour repérer les bombes. Chaque élément est constitué de deux octets contenant l'adresse de la bombe. La création de ces bombes est assurée par le sous programme de tir TBA qui effectue

une lecture du clavier pour savoir si le joueur appuie sur les touches Q ou W. Dans ce cas, et s'il reste une place dans le tableau, il met dans cette place l'adresse de la proue ou de la poupe du bateau en positionnant le bit 15 de l'adresse à un pour indiquer que la bombe vient d'être lancée. Le programme de tir devra être appelé trois fois moins souvent que le programme de déplacement pour être sûr que les bombes sont toutes lachées sur le même segment graphique.

Le déplacement des bombes est réalisé par le sous programme ABM qui ne présente pas de difficulté tant que les bombes restent dans un même caractère. Lorsqu'on saute de ligne, on commence par effacer le caractère d'où on vient, sauf si la bombe vient d'être lancée. On teste ensuite la case mémoire où on va se placer afin de déterminer si la bombe ne rencontre pas un autre mobile. Si la case est vide ou contient une autre bombe, la première bombe continue sa route jusqu'en bas de l'écran où elle est détruite par la mise à zéro de l'octet de poids fort de la variable associée. Si la case contient une mine, le sous programme ETO est appelé pour indiquer que cette mine est cachée par la bombe. Si la case contient un autre caractère, cela signifie qu'on se trouve en présence d'un sous-marin. Un des sous programmes d'explosion (ESG ou ESD) est alors invoqué pour détruire le sous-marin et augmenter le score du joueur.

3.7.4. Déplacement des mines

Les mines lachées par les sous-marins sont représentées par une étoile qui se déplace verticalement de bas en haut en avançant d'une ligne à chaque fois. Le nombre de mines visibles sur l'écran est limité à dix par l'utilisation d'un tableau ayant dix éléments. Chaque élément est constitué de deux octets contenant l'adresse de la mine. Si le second octet est nul la mine n'existe pas. Si le bit sept de cet octet vaut zéro, la mine se trouve actuellement cachée par un autre objet.

La création des mines s'effectue de façon aléatoire par le sous programme de tir TSM qui utilise P2 comme paramètre lors de l'appel de la fonction RND. La naissance d'une mine est réalisée en initialisant un élément libre du tableau avec l'adresse située au milieu du sous-marin, en gardant le bit 15 de cette adresse à zéro pour indiquer que la mine est actuellement masquée par le sous-marin.

Le déplacement des mines est assuré par le sous programme AMN. Celui-ci commence par effacer les mines précédentes qui ne sont pas cachées par un autre mobile. L'adresse de chaque mine est diminuée de 64 pour passer à la ligne précédente. Si la mine arrive à la surface de la mer elle éclate en produisant une gerbe d'eau constituée de cinq caractères graphiques. Si le bateau se trouve suffisemment près pour être atteint par la mine ou par les éclats de l'explosion, le contrôle est donné au programme BOUM qui détruit le bateau du joueur. Dans le cas contraire, l'octet de poids fort de l'adresse est mis à un pour indiquer au programme AMN qu'il devra effacer les éclats de l'explosion la prochaine fois qu'il sera appelé. Si la mine n'atteint pas la surface, on examine la case mémoire où elle va se poser de façon à s'assurer qu'elle ne se place pas sur un mobile. Si c'est le cas, le bit 7 du second octet de la variable associée est mis à zéro pour indiquer que la mine est cachée. Dans le cas contraire on dessine la mine, et on positionne le bit 7 à un.

3.7.5. Autres sous programmes

Le sous programme DES effectue le dessin initial de l'écran. Il efface l'écran, dessine la mer, le bateau et le message sur la première ligne de l'écran. Le sous programme INIT initialise toutes les variables et les paramètres. Il met à zéro les tableaux utilisés dans les déplacements pour indiquer que tous les mobiles, sauf le bateau, n'existent pas encore.

4

Sonorisation

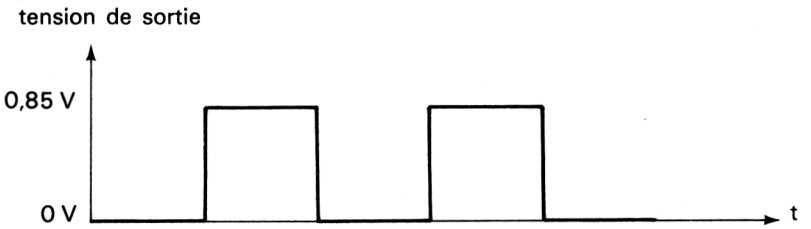
La sonorisation d'un jeu est une opération qui se fait en général à la fin de la mise au point du programme quand celui-ci commence à « tourner » à peu près correctement. Cette opération peut se faire très simplement si le programme a été bien conçu en suivant la méthodologie étudiée dans les chapitres précédents. Il suffit en effet de remplacer certaines boucles d'attente du programme par des appels à des sous programmes de son. Cette phase de sonorisation ne doit pas être négligée parce qu'elle ajoute énormément au programme, un jeu sonore est toujours beaucoup plus attrayant qu'un jeu muet. Il y aura donc intérêt à rechercher par l'expérimentation des sonorités agréables qui puissent imiter les sons produits par les différents personnages.

4.1. PRODUCTION DU SON

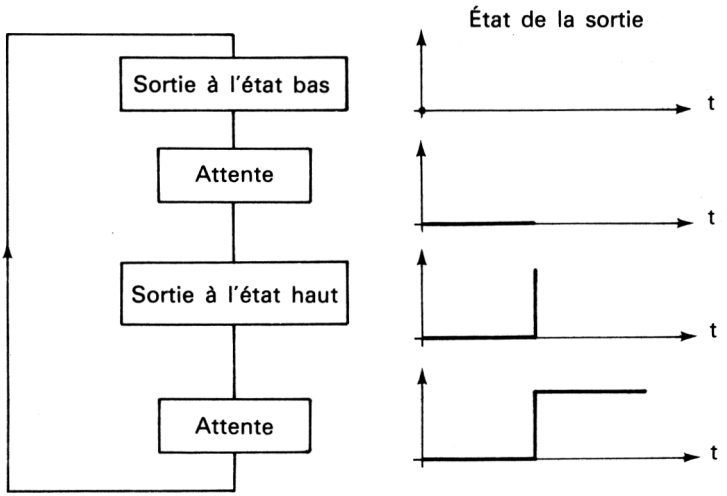
Pour produire du son sur le TRS 80, on emploie la sortie magnétophone utilisée pour sauvegarder des programmes sur cassette. Pour commander cette sortie, il suffit d'effectuer une instruction OUT du Z 80 sur le port d'entrée/sortie portant le numéro 255. Cette sortie peut prendre trois valeurs de tension distinctes, selon l'octet envoyé sur le port 255. La correspondance entre les octets envoyés et les tensions prises est résumée dans le tableau suivant :

Niveau	Tension	Valeur de l'octet à envoyer sur le port 255
haut	0,85 V	2
intermédiaire	0,46 V	0
bas	0 V	1 ou 3

En faisant varier rapidement la tension prise par la sortie magnétophone, on génère un signal électrique qui, une fois relié à un amplificateur muni d'un haut parleur, produira un son. Il est ainsi possible de créer un signal carré dont la fréquence pourra être ajustée afin de produire la note de musique désirée. Ce signal est représenté par le schéma ci-dessous :



Pour générer ce signal on utilisera un programme répondant à l'organigramme suivant :



Pour que le son ne dure pas éternellement comme dans cet organigramme il y aura lieu d'utiliser un deuxième compteur limitant le nombre d'impulsions générées. Il sera ainsi possible de modifier la durée de la note en choisissant le nombre d'impulsions envoyées, et de modifier sa hauteur en diminuant ou en augmentant la durée crée par la boucle d'attente. Plus la boucle d'attente sera grande plus le son sera grave, plus elle sera petite plus le son sera aigu.

```
7A00            00010          ORG     7A00H
7A00 3E01       00020 DEBUT    LD      A,1       ;CODE DU NIVEAU BAS EN SORTIE
7A02 0EC8       00030          LD      C,200     ;COMPTEUR DE DUREE DE LA NOTE
7A04 D3FF       00040 LOOP     OUT     (255),A   ;MODIFICATION DE LA SORTIE
7A06 0696       00050          LD      B,150     ;COMPTEUR DE LA BOUCLE D'ATTENTE
7A08 10FE       00060 WAIT     DJNZ    WAIT      ;BOUCLE D'ATTENTE
7A0A EE03       00070          XOR     3         ;A PASSE DE 1 A 2 OU DE 2 A 1
7A0C 0D         00080          DEC     C         ;DECREMENTER COMPTEUR DUREE
7A0D 20F5       00090          JR      NZ,LOOP   ;SI COMPTEUR NON NUL CONTINUER
7A0F 18FE       00100 LA       JR      LA        ;ARRET
7A00            00110          END     DEBUT
00000 TOTAL ERRORS
```

Dans le programme en assembleur ci-dessus, il sera possible d'évaluer la fréquence de la note générée en calculant la durée produite par la boucle d'attente. Entre deux valeurs de la tension de sortie le microprocesseur utilise un nombre de cycle d'horloge égal à $11 + 7 + 149 * 13 + 8 + 7 + 4 + 12 = 1986$. La fréquence de l'horloge du TRS 80 étant fixée à 1,77 MHz, celle du son généré sera égale à 1,77 MHz/1986 soit 891 Hz (fréquence audible).

Les notes de musique ont une fréquence bien déterminée. Aussi il est possible de calculer le nombre initialisant le compteur de boucle d'attente pour produire la note désirée. Appelons N ce nombre. La fréquence générée est alors donnée par la formule suivante:

$$F = \frac{1{,}77\ 10^6}{46 + (N - 1) * 13} \quad \text{d'où} \quad N = \frac{1}{13}\left(\frac{1{,}77\ 10^6}{F} - 46\right) + 1$$

Étant donné que la durée de la boucle d'attente est variable suivant la note émise, il faudra ajuster le compteur de durée pour générer des notes de même longueur. Soit D_1 et D_2 les valeurs initialisant les compteurs de boucle de durée respectivement pour les fréquences F_1 et F_2, on a la formule suivante:

$$\frac{D_1}{F_1} = \frac{D_2}{F_2}$$

Une fois la durée d'une note choisie, cette formule permettra de déterminer tous les nombres D servant à initialiser les compteurs de boucle. Les valeurs de ces paramètres correspondant à la gamme sont données dans le tableau suivant:

Note	DO	RE	MI	FA	SOL	LA	SI	DO
Fréquence (Hz)	524	588	660	698	784	880	988	1046
Compteur de boucle d'attente	256	229	204	193	171	152	135	128
Compteur de boucle de durée	126	141	158	167	188	211	237	251

Le programme ci-dessous parcourt éternellement la gamme dans un sens puis dans l'autre. Afin d'animer le jeu, vous pourrez utiliser ce programme pour générer un petit air triomphateur à la fin de la partie en modifiant les valeurs des constantes regroupées dans le tableau TAB.

```
7A00            00010           ORG     7A00H
7A00 212A7A     00020 DEBUT     LD      HL,TAB    ;INITIALISATION POINTEUR SUR TAB
                00030 ;PARCOURIR LA GAMME A L'ENDROIT
7A03 1E08       00040 BOUCLE    LD      E,8       ;8 NOTES DANS LA GAMME
7A05 4E         00050 A0        LD      C,(HL)    ;POUR COMPTEUR DE BOUCLE D'ATTENTE
7A06 23         00060           INC     HL        ;INCREMENTER POINTEUR
7A07 56         00070           LD      D,(HL)    ;COMPTEUR DE DUREE
7A08 23         00080           INC     HL        ;INCREMENTER POINTEUR
7A09 CD1D7A     00090           CALL    SON       ;GENERATION D'UNE NOTE
7A0C 1D         00100           DEC     E         ;COMPTEUR NOMBRE DE NOTES
7A0D 20F6       00110           JR      NZ,A0     ;PASSER A LA NOTE SUIVANTE
                00120 ;PARCOURIR LA GAMME A L'ENVERS
7A0F 1E08       00130           LD      E,8       ;8 NOTES DANS LA GAMME
7A11 2B         00140 A1        DEC     HL        ;DECREMENTER POINTEUR
7A12 56         00150           LD      D,(HL)    ;COMPTEUR DE DUREE
7A13 2B         00160           DEC     HL        ;DECREMENTER POINTEUR
7A14 4E         00170           LD      C,(HL)    ;POUR COMPTEUR DE BOUCLE D'ATTENTE
7A15 CD1D7A     00180           CALL    SON       ;GENERATION D'UNE NOTE
7A18 1D         00190           DEC     E         ;COMPTEUR NOMBRE DE NOTES
7A19 20F6       00200           JR      NZ,A1     ;PASSER A LA NOTE SUIVANTE
7A1B 18E6       00210           JR      BOUCLE    ;RECOMMENCER ETERNELLEMENT
                00220 ;*****************************************************
                00230 ;
                00240 ;SOUS PROGRAMME DE SON
                00250 ;
                00260 ;EN ENTREE: D=COMPTEUR DE BOUCLE DE DUREE
                00270 ;             C SERT A INITIALISER LE COMPTEUR DE BOUCLE
                00280 ;               D'ATTENTE
                00290 ;
                00300 ;*****************************************************
7A1D 3E01       00310 SON       LD      A,1       ;CODE DU NIVEAU BAS EN SORTIE
7A1F D3FF       00320 LOOP      OUT     (255),A   ;MODIFICATION DE LA SORTIE
7A21 41         00330           LD      B,C       ;COMPTEUR DE BOUCLE D'ATTENTE
7A22 10FE       00340 WAIT      DJNZ    WAIT      ;BOUCLE D'ATTENTE
7A24 EE03       00350           XOR     3         ;A PASSE DE 1 A 2 OU DE 2 A 1
7A26 15         00360           DEC     D         ;DECREMENTER COMPTEUR DUREE
7A27 20F6       00370           JR      NZ,LOOP   ;SI COMPTEUR NON NUL CONTINUER
7A29 C9         00380           RET               ;RETOUR
                00390 ;
```

```
                00400 ;TABLEAU DES CONSTANTES
                00410 ;
7A2A 00         00420 TAB       DEFB    0           ;ATTENTE DO
7A2B 7E         00430           DEFB    126         ;DUREE DO
7A2C E5         00440           DEFB    229         ;ATTENTE RE
7A2D 8D         00450           DEFB    141         ;DUREE RE
7A2E CC         00460           DEFB    204         ;ATTENTE MI
7A2F 9E         00470           DEFB    158         ;DUREE MI
7A30 C1         00480           DEFB    193         ;ATTENTE FA
7A31 A7         00490           DEFB    167         ;DUREE FA
7A32 AB         00500           DEFB    171         ;ATTENTE SOL
7A33 BC         00510           DEFB    188         ;DUREE SOL
7A34 98         00520           DEFB    152         ;ATTENTE LA
7A35 D3         00530           DEFB    211         ;DUREE LA
7A36 87         00540           DEFB    135         ;ATTENTE SI
7A37 ED         00550           DEFB    237         ;DUREE SI
7A38 80         00560           DEFB    128         ;ATTENTE DO
7A39 FB         00570           DEFB    251         ;DUREE DO
7A00            00580           END     DEBUT
00000 TOTAL ERRORS
```

Pour obtenir des sonorités plus complexes, il faudra moduler la durée produite par la boucle d'attente en modifiant sa longueur au cours du temps. La modification la plus simple consiste à décrémenter la variable initialisant le compteur de boucle d'attente chaque fois que la sortie change d'état. On obtient de la sorte un son très intéressant qui est utilisé dans de nombreux jeux.

```
7A00            00010           ORG     7A00H
7A00 3E01       00020 DEBUT     LD      A,1         ;CODE DU NIVEAU BAS EN SORTIE
7A02 0EC8       00030           LD      C,200       ;COMPTEUR DE DUREE DE LA NOTE
7A04 D3FF       00040 LOOP      OUT     (255),A     ;MODIFICATION DE LA SORTIE
7A06 41         00050           LD      B,C         ;COMPTEUR DE LA BOUCLE D'ATTENTE
7A07 10FE       00060 WAIT      DJNZ    WAIT        ;BOUCLE D'ATTENTE
7A09 EE03       00070           XOR     3           ;A PASSE DE 1 A 2 OU DE 2 A 1
7A0B 0D         00080           DEC     C           ;DECREMENTER COMPTEUR DUREE
7A0C 20F6       00090           JR      NZ,LOOP     ;SI COMPTEUR NON NUL CONTINUER
7A0E 18FE       00100 LA        JR      LA          ;ARRET
7A00            00110           END     DEBUT
00000 TOTAL ERRORS
```

En modifiant la durée de la boucle d'attente par des procédés différents vous obtiendrez des sonorités plus ou moins bonnes. Expérimentez les et retenez celles qui vous semblent les meilleures. En combinant ces sonorités élémentaires vous pourrez parfois en obtenir de bien meilleures.

4.2. IMPLANTATION DU SON DANS LE PROGRAMME DE JEU

Les diverses sonorités générées correspondent en général à un événement particulier qui se produit au cours du jeu (avance d'un ennemi, tir, explosion, bonus). Aussi elles seront commandées par la partie du programme créant cet événement et elles essayeront d'imiter au mieux le bruit produit par celui-ci. Un sous programme sera construit pour chaque sonorité. Il sera appelé soit directement au moment où se produit l'événement (explosion, bonus) soit juste après, à la place de la boucle d'attente générale, pour ne pas ralentir le déroulement du jeu. Aussi on devra inclure dans la boucle principale du jeu une procédure déterminant si on doit exécuter un des sous programmes de son ou bien la boucle d'attente, celle-ci étant prise par défaut. Une variable poura être utilisée pour stocker l'adresse du sous programme où on devra se brancher. Cette variable sera initialisée avec l'adresse du sous programme d'attente au début de la boucle principale. Ensuite elle sera éventuellement modifiée par les programmes de déplacement en respectant des règles de priorité (certains sons étant plus prioritaires que d'autres).

La boucle principale aura l'allure suivante:

```
00010 LOOP    LD      HL,WAIT   ;ADRESSE DU SOUS PROGRAMME  D'ATTENTE
00020         LD      (ADR),HL  ;INITIALISER LA VARIABLE AVEC CETTE ADRESSE
00030         .
00040         .
00050         DEPLACEMENT DES PERSONNAGES
00060         MODIFICATION EVENTUELLE DE LA VARIABLE ADR
00070         .
00080         .
00090         LD      DE,SUIT   ;ADRESSE DE RETOUR
00100         PUSH    DE        ;EMPILER CETTE ADRESSE, L'EXECUTION DE RET
00110                           ;RAMENERA A L'ADRESSE SUIT
00120         LD      HL,(ADR)  ;ADRESSE DU SOUS PROGRAMME
00130         JP      (HL)      ;APPEL DU SOUS PROGRAMME
00140 SUIT    JP      LOOP      ;BOUCLER
00150 ;
00160 ADR     DEFS    2         ;VARIABLE
```

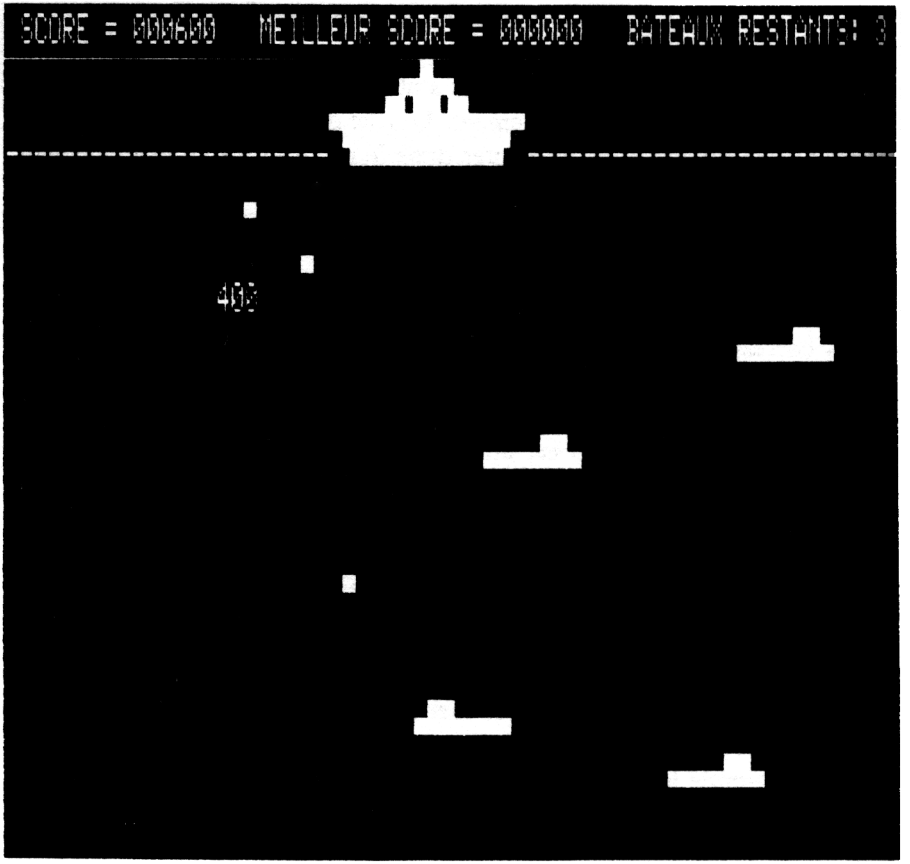
Si le nombre de sonorités est important, il pourra s'avérer nécessaire de partitionner la boucle d'attente en plusieurs sous boucles de durée plus faible auxquelles on associera une variable. Il sera ainsi possible d'émettre plusieurs sons simultanément à chaque passage dans la boucle principale du jeu.

4.3. APPLICATION A NOTRE EXEMPLE

La détermination des sonorités à utiliser se fait par expérimentation directe sur le jeu en fonctionnement. Le rôle du programmeur consiste à trouver des sons et à les implanter dans le programme en évaluant leur durée par le calcul, de façon à ne pas perturber outre mesure le déroulement du jeu. Selon le résultat obtenu il pourra modifier certaines sonorités, en ajouter ou en détruire d'autres.

Dans notre exemple nous avons utilisé six sous programmes de sons correspondant à divers événements. Le sous programme SNO est appelé lors de l'explosion d'un sous-marin pour simuler le bruit qui en résulte. De même SN1 simule le bruit de l'explosion du bateau du joueur. Le sous programme SN8 est utilisé de façon répétitive pour annoncer, par une sonnerie, le bateau supplémentaire gagné par le joueur et durant la présentation pour produire une sonorité amusante. Ces trois sous programmes utilisent à peu de chose près un procédé analogue à celui employé dans le dernier programme de son donné dans ce chapitre. Ils sont d'autre part appelés directement par le processus qui les commande.

La boucle d'attente principale du jeu a été divisée en deux sous boucles de durées égales. La première est utilisée soit pour produire la boucle d'attente proprement dite, soit pour appeler le sous programme SN5 qui simule le tir du bateau. Cette sonorité est commandée à partir du sous programme de tir TBA et est exécutée à la fin de la boucle principale du jeu. La seconde partie de la boucle d'attente est utilisée soit pour produire une attente, soit pour appeler le sous programme SN4 générant une note aigüe, soit pour appeler SN7. Le sous programme SN4 est employé lors du tir d'un sous-marin (TSM), le sous programme SN7 est commandé quand une mine éclate à la surface de la mer. Ces trois sous programmes de sons ont été calculés de façon à ce que leur durée soit identique à celle fournie par chaque sous boucle d'attente.


SCORE = 000600
MEILLEUR SCORE = 000000
BATEAUX RESTANTS: 3
400

5

Présentation et amélioration du jeu

Vous venez de réaliser un programme de jeu complet et vous pouvez vous estimer satisfait. Cependant il faut garder présent à l'esprit que vous ne serez pas le seul à y jouer. Aussi il est bon, à ce stade, d'y ajouter toutes les floritures et les améliorations diverses qui en font tout le charme. Il faudra en particulier inclure une présentation originale, prévoir une progression de la difficulté et un mode de fonctionnement automatique (démonstration). Toutes ces améliorations décrites dans ce chapitre ne présentent pas de difficultés de réalisation énormes ; aussi nous ne nous attarderons pas trop sur leur programmation. Nous citerons au contraire des exemples d'améliorations possibles en nous inspirant des programmes commercialisés. Néanmoins, il faut insister sur le rôle primordial de l'imagination dans cette phase du jeu. Un programme présentant des caractéristiques nouvelles sera toujours plus attrayant qu'un programme reprenant des extensions classiques.

5.1. PRÉSENTATION

La présentation est quasiment indépendante du programme principal, aussi elle pourra être créée et mise au point indépendemment de celui-ci. Quand elle sera achevée, elle pourra être regroupée avec le jeu. Le programme de présentation commencera par annoncer le nom du jeu en utilisant si possible des grosses lettres dessinées avec des caractères

graphiques. Au lieu d'apparaître instantanément sur l'écran, ce nom pourra être dessiné par un dispositif rappelant certains aspects du jeu. Il pourra par exemple être dessiné par un rayon Laser, par les éclats d'une explosion ou par des envahisseurs descendant d'un vaisseau spatial. A vous d'être assez imaginatif pour trouver un moyen de faire apparaître ce nom sur l'écran.

Il sera bon d'inclure le mode d'emploi du jeu dans le programme lui-même. Ce mode d'emploi sera affiché sur demande de l'utilisateur (frappe d'une touche spéciale). Il ne devra pas être trop long pour ne pas décourager certains joueurs pressés ; aussi il est conseillé de le faire tenir sur un seul écran en résumant l'essentiel des règles du jeu. Il est également souhaitable de rendre mobile certaines parties du mode d'emploi afin de le rendre plus attractif.

Dans un programme de présentation complet, il sera intéressant de gérer une table classée contenant les dix meilleurs scores réalisés avec en vis-à-vis le nom de leur auteur. Cette table sera affichée en permanence lors de la présentation à des fins d'émulation. Lorsqu'un nouveau score aura été effectué, le programme demandera le nom du joueur en utilisant un sous programme d'entrée de données analogue au sous programme CUR du chapitre 2. Si vous possédez des lecteurs de disques il sera possible de sauvegarder les dix meilleurs scores sur disquette. Pour cela vous pourrez employer les sous programmes d'entrée/ sortie classiques décrits dans le manuel du système d'exploitation (DOS). Les sous programmes qui vous seront utiles sont au nombre de six et sont implantés à des adresses identiques sur tous les systèmes d'exploitation des TRS 80 modèle I ou III. Il s'agit de :

INIT	4420 H	Création du fichier
OPEN	4424 H	Ouverture du fichier
POSN	4442 H	Positionnement sur un secteur particulier
READ	4436 H	Lecture d'un secteur
WRITE	4439 H	Écriture d'un secteur
CLOSE	4428 H	Fermeture du fichier

La présentation constitue le niveau de base d'un programme. Au chargement du programme c'est elle qui est activée en premier, c'est à

partir d'elle qu'on demande le commencement du jeu et c'est vers elle qu'on revient une fois la partie terminée. Il faudra donc prévoir la possibilité de commander le début du jeu par appui sur une simple touche par exemple. D'autre part il sera bon de pouvoir arrêter la partie à tout moment pour revenir à la présentation. Cet arrêt sera commandé par l'appui de deux touches simultanées pour plus de sécurité (par exemple BREAK + CLEAR). La lecture de ces touches pourra être faite à chaque passage dans la boucle principale du jeu et raménera à la présentation dans le cas où elles seraient enfoncées. Dans certains programmes il faudra même faire ces lectures dans des parties qui monopolisent l'ordinateur assez longtemps sans passer par la boucle principale (par exemple l'explosion du vaisseau piloté par le joueur).

5.2. AMÉLIORATIONS

Pour que le jeu conserve tout son intérêt et qu'il ne soit pas délaissé au bout de quelques jours, il faut absolument prévoir une progression de la difficulté au cours de la partie. En effet, si la difficulté reste constante les joueurs sauront déjouer tous les pièges au bout de quelques heures et pourront faire durer la partie quasiment éternellement. Il n'y aura donc plus aucun intérêt à établir un record puisqu'il suffira d'être patient. Par conséquent il est nécessaire de faire progresser la difficulté du jeu au fur et à mesure de la partie pour que le joueur reste constamment motivé. Il sera même souhaitable que cette progression soit visible afin que le joueur s'aperçoive des étapes à atteindre. Au bout d'un certain temps ou au bout d'un certain nombre de points, on pourra faire apparaître un nouveau piège (vaisseaux ennemis supplémentaires plus féroces, ennemis inertes qui deviennent très dangereux) ou un nouveau mode de fonctionnement (vulnérabilité plus grande du vaisseau du joueur, necessité d'employer deux balles au lieu d'une pour tuer un envahisseur). Ces extensions supplémentaires pourront provoquer du même coup un accroissement des points marqués par le joueur à la plus grande joie de celui-ci et afin de garder un jeu équitable. En parallèle à ces extensions brutales, il sera bon également d'effectuer une progression de la difficulté plus lente et plus continue. Ceci pourra se faire en diminuant progressivement la durée de la boucle d'attente principale ou en modifiant la valeur de certains paramètres qui influent sur le nombre d'ennemis ou sur le nombre de bombes envoyés contre le joueur.

Il sera intéressant de prévoir un mode automatique pour montrer le fonctionnement du jeu. Ce mode pourra être activé sur demande de l'utilisateur ou automatiquement pendant la phase de présentation si le joueur tarde trop à taper sur la touche faisant démarrer la partie. Le programme se déroulera alors sans l'intervention du joueur humain. Le vaisseau que celui-ci aurait du piloter sera déplacé automatiquement soit de façon aléatoire, ce qui est le plus facile à réaliser, soit en utilisant un algorithme qui le fera gagner. Pendant cette phase de démonstration il faudra effectuer une lecture du clavier dans la boucle principale du jeu pour tester si l'utilisateur tape sur le clavier. Si tel est le cas on reviendra immédiatement à la présentation.

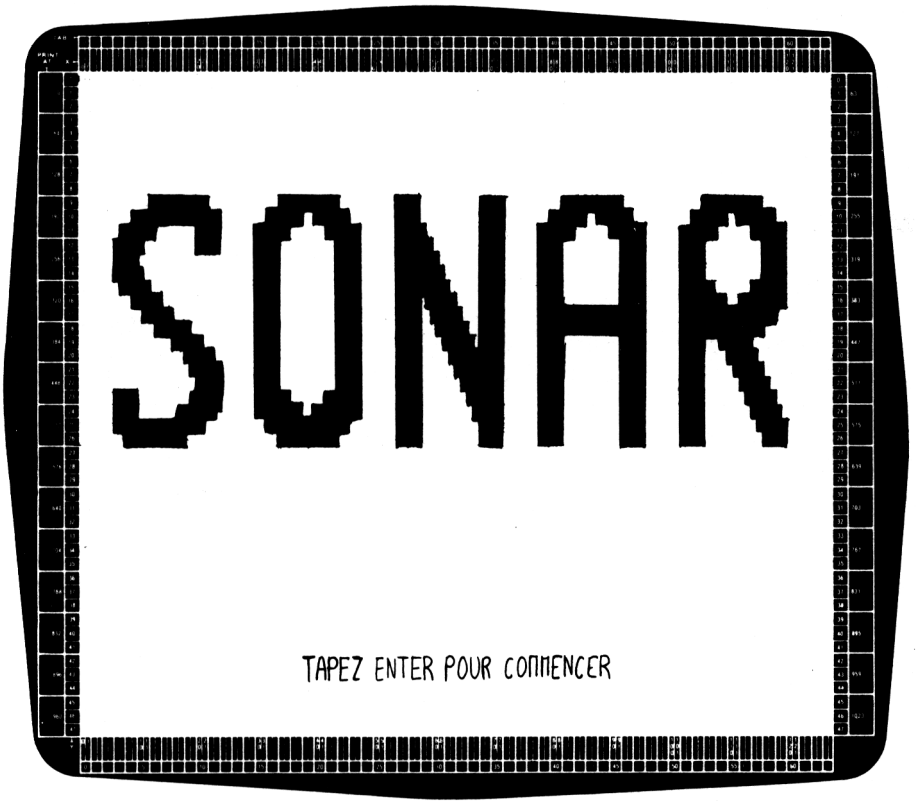
5.3. APPLICATION A NOTRE EXEMPLE

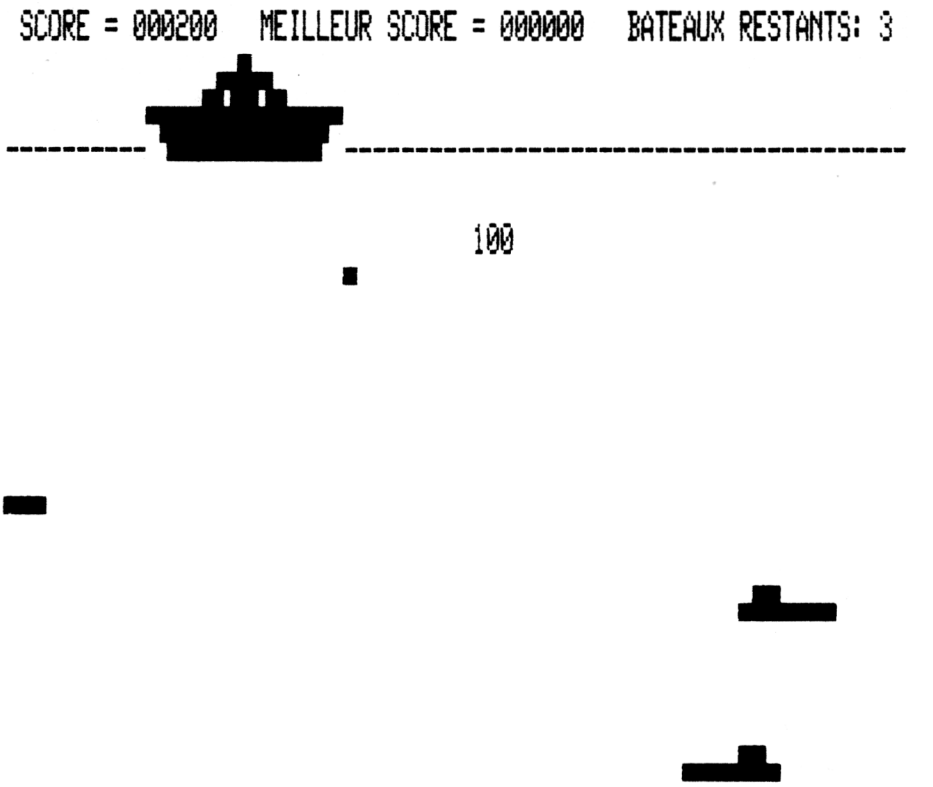
Étant donné que la présentation n'est pas très difficile à réaliser et pour ne pas trop encombrer cet ouvrage de listings, nous nous sommes bornés à réaliser une présentation toute simple. Elle fait apparaître le nom du jeu (SONAR) en grosses lettres sur l'écran. Au-dessous du nom un petit message invite le joueur à taper ENTER pour commencer la partie. Vous trouverez à la page suivante le dessin de l'écran sur feuille spéciale. Le listing commenté du programme de démonstration se trouve dans l'annexe 3.

Nous avons ajouté dans le programme quelques instructions pour obtenir une progression de la difficulté. Pour cela, nous utilisons une variable notée CNT qui compte le nombre de passages dans la boucle principale. Au bout d'un certain nombre de passages, on augmente la difficulté en modifiant les paramètres P0 et P2. Le fait de diminuer P0 provoque un accroissement du nombre de sous-marins sur l'écran. De même la diminution de P2 provoque l'augmentation du nombre de mines. Ainsi, à chaque fois que la variable CNT augmente d'une certaine quantité, on diminue un des paramètres et le jeu devient plus difficile.

Nous avons inclus dans le programme un mode démonstration durant lequel le jeu se déroule de façon aléatoire sans l'intervention du joueur humain. Ce mode est activé automatiquement à partir de la présentation lorsque le joueur tarde à envoyer une commande. On positionne à ce moment-là un indicateur stocké dans la variable JEA. Selon

l'état de cet indicateur, les sous programmes TBA et ABA feront des lectures du clavier pour exécuter la commande du joueur ou bien commanderont le bateau de façon aléatoire en appelant la fonction RND. Durant le mode démonstration on effectuera une lecture du clavier à intervalles réguliers, de façon à voir si le joueur tape sur le clavier. Dans ce cas, on reviendra immédiatement à la présentation.




SCORE = 000200 MEILLEUR SCORE = 000000 BATEAUX RESTANTS: 3
100

6
Méthodologie de réalisation et de mise au point du jeu

Les conseils donnés dans ce chapitre sont le fruit d'une expérience personnelle de la réalisation des programmes de jeu d'animation en assembleur. Ils visent à réduire le temps passé à faire le jeu, en diminuant essentiellement le temps de mise au point ce qui accroit du même coup la fiabilité du programme. Certains de ces conseils vous paraîtront évidents si vous les utilisez déjà, d'autres au contraire feront découvrir une nouvelle méthode qui vous fera gagner du temps.

6.1. MATÉRIEL NÉCESSAIRE ET MATÉRIEL SOUHAITABLE

Les versions objets (langage machine) de la grande majorité des programmes de jeu peuvent s'exécuter sur un TRS 80 modèle I ou III disposant de 16 koctets ou parfois de 4 koctets de mémoire. Par contre la version source écrite en assembleur occupe environ sept fois plus de place. Ainsi pour contenir en entier la version source d'un programme de 4 koctets en langage machine, il faut posséder au moins 28 koctets. A cela, il faut ajouter l'espace nécessaire pour loger l'Éditeur/Assembleur, soit environ 9 ko, plus la place prise par la table des symboles qui fait couramment 2 ko. Nous arrivons ainsi à une taille memoire minimum de

39 koctets qui nécessite l'emploi d'un TRS 80 48 ko. Il est possible néanmoins de produire un jeu dont la source ne tient pas en entier dans l'espace mémoire disponible en le fragmentant en plusieurs parties. Il faudra prévoir cette éventualité au moment de l'écriture du programme, en lui donnant une structure très modulaire. Chaque partie sera assemblée à un emplacement fixe et il faudra limiter au maximum les dépendances entre les différents modules. Ceux-ci contiendront en tête la valeur des étiquettes contenues dans les autres parties et qui sont nécessaires pour l'assemblage. Elles seront définies par des directives EQU qui pourront être facilement modifiées quand un des modules aura été changé. On aura intérêt à mettre au point les différentes parties séparément les unes des autres. Ensuite pour obtenir le programme final on assemblera les modules les uns à la suite des autres en notant la valeur des étiquettes utiles pour les autres modules. Ces valeurs viendront se placer dans les directives EQU afin de produire un assemblage correct. Une fois tous les modules assemblés, on les chargera tous en mémoire et on pourra commencer l'exécution. La sauvegarde de l'ensemble pourra se faire par un utilitaire permettant la mémorisation sur un support magnétique de blocs de mémoire non contigus. De tels utilitaires sont actuellement disponibles sur le marché, et vous ne serez pas obligés de les construire vous même.

Cette procédure est, comme vous vous en doutez, très fastidieuse et éprouvante pour les nerfs surtout si vous utilisez des cassettes comme mémoire de masse. Pratiquement il est difficile de dépasser plus de trois modules, ce qui permet quand même de réaliser des jeux dont la version exécutable tient sur 3 ou 4 koctets, avec un TRS 80 16 k. Cela étant assez pénible nous pensons que l'extension la plus prioritaire pour réaliser des programmes assez importants en assembleur, est l'adjonction de mémoire supplémentaire. Aussi si vous en avez l'occasion n'hésitez pas à faire cette dépense qui vous fera gagner un temps précieux, et par là même augmentera la sureté de vos programmes.

Un programme de jeu assez sophistiqué, ayant une bonne présentation, comporte environ 2 à 3 000 instructions dont 1 000 pour la présentation. Pour écrire ou lire ces instructions sur cassette, il faut compter un temps d'attente d'environ 1/4 d'heure. En comptant le temps nécessaire pour charger l'assembleur, il faut environ attendre vingt minutes entre le moment où le TRS est hors-tension et l'instant où le programme est

chargé et prêt à être modifié. Aussi, si vous utilisez des cassettes pour stocker vos programmes il faudra agir de façon méthodique pour minimiser les transferts de données. On aura intérêt en particulier à corriger en une seule fois toutes les erreurs qu'on aura détecté lors de l'exécution, et à écrire le programme en entier d'une seule fois plutôt que de l'écrire petit à petit. Au contraire, si vous possédez des unités de disquette, vous obtiendrez une plus grande souplesse d'utilisation ce qui facilitera la mise au point de vos programmes. Le temps d'attente au chargement sera suffisemment réduit pour que cela ne soit pas génant. Il sera alors relativement rapide de faire une petite modification du programme. Il existe d'autre part des utilitaires fonctionnant sur disque et pouvant effectuer l'assemblage d'un programme qui se trouve soit entièrement sur disque, soit partiellement en mémoire et sur disque. La taille des versions sources n'est alors limitée que par la capacité des disquettes (environ 100 ko) et par la taille de la table des symboles qui se trouve en mémoire centrale (32 ko sont disponibles pour cette table dans un TRS 80 48 k). Il est alors possible de construire des programmes beaucoup plus évolués contenant un jeu à plusieurs facettes qui sera sans doute beaucoup plus attrayant. Ce que nous verrons de dire dans ce paragraphe montre les avantages séduisants apportés par les lecteurs de disques, mais ils ne sont pas indispensables, à notre avis, pour peu que l'on soit méthodique et organisé. Il est également possible de faire un bon programme de jeux dans un temps relativement court en n'utilisant que des cassettes et en n'étant pas trop impatient.

Une imprimante n'est pas indispensable si vous avez pris soin d'écrire le programme en entier sur une feuille de papier avant de commencer à le taper. Si vous êtes déjà familier au langage machine, vous ne devriez pas faire trop d'erreurs de programmation, et il vous sera alors possible de reporter les corrections effectuées sur votre feuille initiale de façon à avoir toujours sous les yeux la bonne version du programme. L'imprimante peut s'avérer utile si vous êtes obligé d'interrompre la mise au point du jeu pendant une période assez importante, où si, étant novice dans la programmation en assembleur, vous avez un pourcentage d'erreurs trop important pour pouvoir faire les corrections sur votre listing initial.

En résumé, nous pensons que le système minimum nécessaire pour pouvoir facilement réaliser un programme de jeu en assembleur est un

TRS 80 avec 48 ko de mémoire. L'extension la plus utile est le lecteur de disques et ensuite l'imprimante.

6.2. CONSEILS DE RÉALISATION

Il est fortement conseillé d'écrire tout d'abord le jeu en entier (en omettant éventuellement la présentation) sur des feuilles de papier avant de commencer à taper quoi que ce soit sur le clavier de votre ordinateur. Même si la tentation est forte de s'assoir devant son TRS 80, il faut à tout prix éviter cela. L'expérience prouve en effet qu'on perd beaucoup plus de temps en agissant ainsi parce que le programme formé est alors moins bien structuré. Le nombre de modifications à apporter est beaucoup plus grand, le nombre de transferts sur support magnétique augmente de façon importante et cela vous ralentira considérablement si vous utilisez des cassettes. Vous n'aurez d'autre part pas en tête un plan d'ensemble du programme, ce qui vous amenera à faire des listings successifs pour vous "en sortir", et si vous ne possédez pas d'imprimante cela vous sera d'autant plus difficile. Au contraire en écrivant au départ tout le jeu sur papier, on a une bien meilleure vue d'ensemble du programme. Il est également recommandé de vérifier soigneusement à la main le fonctionnement du programme avant de commencer à taper sur le TRS 80. Un programme ainsi réalisé sera certainement beaucoup plus fiable, le nombre de modifications apportées par des erreurs devient très faible. Par conséquent le temps de mise au point se trouve diminué de façon importante et bien que vous ayez passé plus de temps à écrire le jeu sur papier vous gagnez dans l'ensemble du temps. Cela est d'autant plus sensible que vos moyens matériels sont plus faibles.

Après ces quelques conseils sur l'écriture des programmes interressons-nous aux assembleurs. Il existe actuellement deux assembleurs disponibles sur cassette. Le plus ancien a pour nom ÉDITEUR/ASSEMBLEUR et est distribué par Radio-Shack. Il a des caractéristiques modestes mais suffisantes dans la plupart des cas. Il a l'avantage d'être relativement petit, ce qui autorise l'assemblage d'un programme source de 7 ko environ sur un TRS 80 16 ko, et d'être très bien documenté puisqu'il est fourni avec un classeur contenant la liste détaillée des instructions du Z 80. Le plus récent se nomme

ÉDITEUR/ASSEMBLEUR PLUS et est distribué par Microsoft. Il contient, en plus des caractéristiques du précédent assembleur, de nouvelles fonctions d'édition (déplacement de lignes, substitution de chaînes de caractères, ...) et d'assemblage (assemblage en mémoire, assemblage conditionnel, Macros, ...). Un utilitaire de mise au point nommé Z BUG est intégré à l'assembleur. Il permet l'exécution immédiate des programmes assemblés en mémoire sans passer par l'intermédiaire des cassettes, ce qui amène un gain de temps appréciable. Cependant il reste quelques "bugs" liés à cet utilitaire ce qui est vraiment dommage. Des versions futures devraient annuler ces erreurs. Voici donc un assembleur performant pour un micro-ordinateur mais qui a le défaut d'être assez gros, ce problème étant en partie résolu par la possibilité de supprimer des portions du programme.

Trois assembleurs sont disponibles sur disque. Le premier fait partie intégrante du NEWDOS 80 distribué par Apparat. C'est exactement l'assembleur de Radio-Shack auquel on a ajouté des instructions permettant le transfert de fichiers sources et objets sur disque. Le second distribué par Radio-Shack se nomme également ÉDITEUR/ASSEMBLEUR. On retrouve chez celui-ci les caractéristiques des assembleurs implantés sur des machines plus importantes. Aussi il emploie des méthodes qui ne sont peut-être pas très adaptées pour un micro-ordinateur de bas de gamme. En particulier, il utilise les lecteurs de disques ordinaires du TRS 80 comme de puissants lecteurs de disques durs, ce qui a pour effet de diminuer la souplesse d'édition des programmes. Néanmoins cet assembleur est très puissant, il possède d'importantes directives d'assemblages inexistantes sur beaucoup de micro-ordinateurs. Il permet également l'assemblage de fichiers sur disques et le relogement des programmes grâce à son éditeur de liens. Le dernier assembleur sur disque s'appelle EDAS et est distribué par Misosys. Il offre la particularité intéressante de permettre l'assemblage d'un fichier se trouvant en partie en mémoire et en partie sur disque, ce qui autorise l'assemblage de sources énormes, sans perdre la souplesse d'édition des assembleurs classiques. Les directives d'assemblage sont intéressantes mais pas aussi complètes que celle du précédent assembleur. D'autre part des erreurs de jeunesse viennent diminuer la fiabilité de ce logiciel. Espérons qu'une version future, supprimera ces "bugs" et ajoutera ou améliorera certaines instructions d'assemblage.

6.3. CONSEILS DE MISE AU POINT

Pour la mise au point du programme, il est bon de posséder un "debugger" puissant (outil d'aide à la mise au point), ce qui n'est pas un achat trop important. Étant donné que le nombre de "debugger" sur le marché est actuellement très grand et que de nouveaux outils sortent tous les ans, nous nous bornerons à citer les caractéristiques essentielles de ces logiciels en indiquant celles qui sont le plus utile. En général les "debugger" ont un minimum de performances qui sont : visualisation et modification des registres internes du Z 80 et de la mémoire en notation hexadécimale, pose d'un point d'arrêt dans le programme, lancement de l'exécution à partir d'une adresse absolue. Ces fonctions sont en principe suffisantes pour faire de la mise au point, mais en pratique il est agréable de posséder des commandes plus puissantes. Aussi un bon "debugger" devra posséder un mode pas à pas permettant d'exécuter les instructions du Z 80 une par une. Ce mode est très utile pour trouver des erreurs incompréhensibles ou pour débuter la programmation en assembleur. Il permet en effet de visualiser à l'écran le fonctionnement de chaque instruction du Z 80 ce qui permet la compréhension des principes de base du microprocesseur. Il est également agréable de disposer d'un programme qui n'interfère pas avec les ROM du basic et qui soit relogeable, c'est-à-dire qui puisse être implanté en mémoire à partir d'une adresse fixée par l'utilisateur afin de ne pas perturber le programme de ce dernier. Une caractéristique intéressante d'un "debugger" pour une application graphique est le fait que celui-ci n'efface pas l'écran quand il est activé. Il est alors possible de reprendre l'exécution après un arrêt sur un point d'arrêt en conservant l'écran intact. Si vous ne connaissez pas les codes des instructions du Z 80 par cœur, il est bon de disposer d'une commande de désassemblage, indiquant en clair l'instruction qui va être exécutée. Enfin des commandes permettant la visualisation de la mémoire en ASCII, l'écriture et la lecture de données en mémoire vers une cassette sont parfois bien utiles.

Si vous débutez en assembleur vous commetrez un nombre d'erreur de programmation assez important, ces erreurs pourront avoir pour effet de détruire votre programme en mémoire et de bloquer le TRS 80. Au lieu de commencer à faire exécuter d'une seule traite votre programme, vous aurez intérêt à le faire exécuter par groupe d'instructions, en posant

des points d'arrêt à la fin de chaque groupe ou en faisant de l'exécution pas à pas. Il sera bon en particulier de poser des points d'arrêt sur les instructions de branchement et aux endroits où il risque de se produire des problèmes. Si vous êtes rompu aux techniques de programmation en langage machine, vous aurez au contraire intérêt à faire tout d'abord exécuter le jeu en entier et à noter les erreurs qui se sont produites. Avec un peu d'habitude vous pourrez corriger la plupart de ces erreurs sans avoir besoin d'un "debugger". Celui-ci sera nécessaire pour comprendre les erreurs résultant d'une mauvaise analyse du jeu. Il faudra détecter l'endroit où se produit l'erreur afin de savoir d'où elle provient et de pouvoir la corriger. Ce sont bien là les problèmes les plus ardus de ce genre de programme et ils risquent de décourager certaines personnes. Il faut cependant s'accrocher et passer le temps nécessaire pour les comprendre et les résoudre. La principale difficulté des jeux d'animation est constituée des tests à effectuer lors du déplacement des mobiles sur l'écran. Il arrive fréquemment qu'on oublie certains tests qui n'ont pas été jugés nécessaires lors de l'analyse. Ces oublis conduisent à un fonctionnement érroné du jeu dans des cas particuliers. Donnons un exemple usuel d'un tel disfonctionnement:

Supposons qu'un vaisseau se déplace horizontalement sur l'écran et qu'un missile se déplaçant verticalement aille à sa rencontre. Le missile détecte à chaque déplacement s'il ne va pas percuter le vaisseau, ce dernier avance sans s'occuper de rien. A l'instant t le missile frôle le vaisseau sans le toucher, ce qui donne la figure suivante:

Le vaisseau effectuant un déplacement élémentaire vers la droite vient écraser le missible ce qui donne:

à l'instant t + 1 le missile continue son chemin sans avoir décelé l'impact et efface par conséquent la case d'où il venait, c'est-à-dire la case contenant la tête du vaisseau.

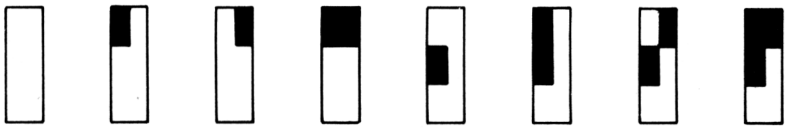
Le vaisseau se trouve ainsi partiellement effacé et cela se verra d'autant plus que son déplacement sera lent.

Certains programmes de jeu commercialisé contiennent des erreurs de ce type et il est à déplorer que leur auteur n'ait pas jugé bon de les corriger par souci de simplification.

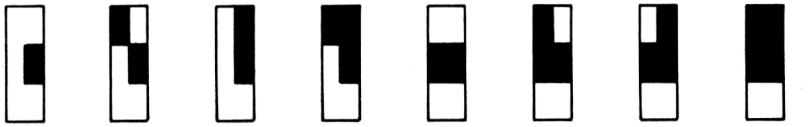
En guise de conclusion insistons sur le fait que les jeux doivent être longuement contrôlés pour être sûr que tous les tests ont été effectués et qu'il n'existe pas un cas de figure où ils puissent être pris à défaut. Ce n'est qu'à ce moment là que le travail du créateur sera terminé. Il pourra alors jouer lui-même avec son programme s'il en a encore le courage et en faire profiter son entourage.

ANNEXE 1

Code des caractères graphiques



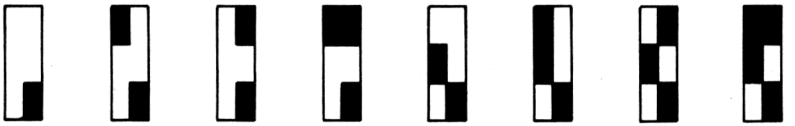
80H 128 | 81H 129 | 82H 130 | 83H 131 | 84H 132 | 85H 133 | 86H 134 | 87H 135



88H 136 | 89H 137 | 8AH 138 | 8BH 139 | 8CH 140 | 8DH 141 | 8EH 142 | 8FH 143

90H 144 | 91H 145 | 92H 146 | 93H 147 | 94H 148 | 95H 149 | 96H 150 | 97H 151

98H 152 | 99H 153 | 9AH 154 | 9BH 155 | 9CH 156 | 9DH 157 | 9EH 158 | 9FH 159



A0H 160 | A1H 161 | A2H 162 | A3H 163 | A4H 164 | A5H 165 | A6H 166 | A7H 167

A8H 168 | A9H 169 | AAH 170 | ABH 171 | ACH 172 | ADH 173 | AEH 174 | AFH 175

B0H 176 | B1H 177 | B2H 178 | B3H 179 | B4H 180 | B5H 181 | B6H 182 | B7H 183

B8H 184 | B9H 185 | BAH 186 | BBH 187 | BCH 188 | BDH 189 | BEH 190 | BFH 191

ANNEXE 2

Structure du clavier

Chaque touche enfoncée du clavier à pour effet de positionner à "1" un des bits d'une des huit mémoires servant à réaliser l'interface entre le clavier et l'unité centrale. Pour connaître l'état du clavier il suffira donc de lire la valeur contenue dans ces mémoires. La correspondance entre les bits et les touches enfoncées est résumée dans le tableau ci-dessous :

Adresse		*Valeur lue*							
Décimal	Hexadécimal	128 bit_7	64 bit_6	32 bit_5	16 bit_4	8 bit_3	4 bit_2	2 bit_1	1 bit_0
14 337	3 801	G	F	E	D	C	B	A	@
14 338	3 802	O	N	M	L	K	J	I	H
14 340	3 804	W	V	U	T	S	R	Q	P
14 344	3 808						Z	Y	X
14 352	3 810	' 7	& 6	% 5	$ 4	# 3	" 2	! 1	 0
14 368	3 820	? /	> .	= –	< ,	+ ;	* :	) 9	(8
14 400	3 840	SPACE	→	←	↓	↑	BREAK	CLEAR	ENTER
14 464	3 880								SHIFT

ANNEXE 3

SONAR

```
4400            00010           ORG     4400H     ;IMPLANTATION POUR 4K
                00020                             ;POUR DISQUE METTRE 7000H
4F80            00030 PILE      EQU     $+0B80H   ;PILE DE TRAVAIL
3C00            00040 DB        EQU     3C00H     ;DEBUT MEMOIRE ECRAN
0005            00050 NBM       EQU     5         ;NOMBRE DE BOMBES
0005            00060 NMN       EQU     5         ;NOMBRE DE MINES
                00070 ;
                00080 ;******************** D E B U T ***************************
                00090 ;
4400 F3         00100 DEBUT     DI                ;
4401 31804F     00110           LD      SP,PILE   ;INITIALISER PILE
4404 210000     00120           LD      HL,0      ;
4407 22CE4D     00130           LD      (MC),HL   ;MISE A ZERO DU MEILLEUR SCORE
440A CDD549     00140           CALL    RANDOM    ;INITIALISER GENERATEUR ALEATOIRE
440D CD514A     00150           CALL    PRMN      ;TEST PRESENCE MINUSCULES
                00160 ;
                00170 ;************** P R E S E N T A T I O N ******************
                00180 ;
4410 CDC749     00190 PRES      CALL    CLS       ;EFFACEMENT ECRAN
4413 AF         00200           XOR     A         ;
4414 32CA4D     00210           LD      (JEA),A   ;MODE ORDINAIRE PAR DEFAUT
4417 21084D     00220           LD      HL,TAP    ;ADR. MESSAGE "TAPEZ ENTER..."
441A 11933F     00230           LD      DE,DB+915;
441D CD614A     00240           CALL    MES       ;AFFICHAGE DU MESSAGE
4420 11C33C     00250           LD      DE,DB+195;IMPLANTATION DU NOM SONAR
4423 218A4B     00260           LD      HL,SONAR  ;ADR. CHAINE GRAPHIQUE
4426 0605       00270           LD      B,5       ;NOMBRE DE LETTRES
4428 C5         00280 N0        PUSH    BC        ;SAUVER CE NOMBRE
4429 3E06       00290           LD      A,6       ;NOMBRE DE LIGNES PAR LETTRE
442B D5         00300           PUSH    DE        ;SAUVER ADR. DEBUT DE LETTRE
442C 010900     00310 N1        LD      BC,9      ;NOMBRE DE CARACTERES PAR LIGNE
442F EDB0       00320           LDIR              ;AFFICHAGE DE 9 CARACTERES
4431 013700     00330           LD      BC,55     ;
4434 EB         00340           EX      DE,HL     ;
4435 09         00350           ADD     HL,BC     ;ADR. LIGNE SUIVANTE
4436 EB         00360           EX      DE,HL     ;
4437 3D         00370           DEC     A         ;COMPTEUR NOMBRE DE LIGNES
4438 20F2       00380           JR      NZ,N1     ;SI NON NUL PASSER A LA SUIVANTE
```

```
443A D1       00390          POP     DE        ;ADR. DEBUT LETTRE
443B 010C00   00400          LD      BC,12     ;
443E EB       00410          EX      DE,HL     ;
443F 09       00420          ADD     HL,BC     ;ADR. LETTRE SUIVANTE
4440 EB       00430          EX      DE,HL     ;
4441 C1       00440          POP     BC        ;NOMBRE DE LETTRES
4442 10E4     00450          DJNZ    N0        ;BOUCLER POUR LES 5 LETTRES
4444 1601     00460          LD      D,1       ;VARIABLE POUR SN8
4446 CD7C4B   00470 T0       CALL    SN8       ;SON
4449 3A4038   00480          LD      A,(3840H);LECTURE CLAVIER
444C FE01     00490          CP      1         ;A-T-ON TAPE ENTER ?
444E 2827     00500          JR      Z,JEU     ;OUI COMMENCER LE JEU
4450 14       00510          INC     D         ;INCREMENTER COMPTEUR
4451 7A       00520          LD      A,D       ;
4452 FE82     00530          CP      130       ;
4454 20F0     00540          JR      NZ,T0     ;SI DIFFERENT DE 130 BOUCLER
4456 CD7C4B   00550 T1       CALL    SN8       ;SON
4459 3A4038   00560          LD      A,(3840H);
445C FE01     00570          CP      1         ;
445E 2817     00580          JR      Z,JEU     ;SI ON A TAPE ENTER COMMENCER
4460 15       00590          DEC     D         ;DECREMENTER COMPTEUR
4461 20F3     00600          JR      NZ,T1     ;SI NON NUL BOUCLER
4463 010090   00610          LD      BC,9000H  ;NOMBRE DE LECTURES
4466 CD2B00   00620 N2       CALL    2BH       ;LECTURE CLAVIER
4469 FE0D     00630          CP      0DH       ;
446B 280A     00640          JR      Z,JEU     ;SI ENTER
446D 0B       00650          DEC     BC        ;
446E 78       00660          LD      A,B       ;
446F B1       00670          OR      C         ;
4470 20F4     00680          JR      NZ,N2     ;SI COMPTEUR NON NUL CONTINUER
4472 3E01     00690          LD      A,1       ;AU BOUT D'UN CERTAIN TEMPS
4474 32CA4D   00700          LD      (JEA),A   ;METTRE FLAG MODE DEMONSTRATION
              00710 ;
              00720 ;************ D E B U T   D U   J E U ******************
              00730 ;
4477 31804F   00740 JEU      LD      SP,PILE   ;INITIALISER PILE
447A CD3845   00750          CALL    INIT      ;INITIALISATIONS DIVERSES
447D CD0945   00760          CALL    DES       ;DESSIN DU DECOR
4480 215F4B   00770 Z0       LD      HL,SN6    ;ADR. S.P. D'ATTENTE
4483 22D04D   00780          LD      (W0),HL   ;INITIALISER W0
4486 22D24D   00790          LD      (W1),HL   ;ET W1
4489 2AD44D   00800          LD      HL,(CNT)  ;COMPTEUR NOMBRE DE BOUCLES
448C 23       00810          INC     HL        ;
448D 22D44D   00820          LD      (CNT),HL  ;INCREMENTE A CHAQUE PASSAGE
4490 3A704D   00830          LD      A,(FPA)   ;
4493 B7       00840          OR      A         ;
4494 2807     00850          JR      Z,M3      ;SI PAS FIN DE PARTIE
4496 116400   00860          LD      DE,100    ;FIN DE PARTIE
4499 DF       00870          RST     18H       ;
449A CA1044   00880          JP      Z,PRES    ;AU BOUT DE 100 BOUCLES ARRETER
449D 11C800   00890 M3       LD      DE,200    ;
44A0 DF       00900          RST     18H       ;
44A1 3823     00910          JR      C,M1      ;SI CNT<200
44A3 200C     00920          JR      NZ,M2     ;SI CNT<>200
44A5 3AC44D   00930          LD      A,(P0)    ;DECREMENTER PARAMETRE P0 DE 10
44A8 D60A     00940          SUB     10        ;
44AA 281A     00950          JR      Z,M1      ;SI NUL NE PAS LE MODIFIER
44AC 32C44D   00960          LD      (P0),A    ;
44AF 1815     00970          JR      M1        ;SUITE
44B1 119001   00980 M2       LD      DE,400    ;
44B4 DF       00990          RST     18H       ;
44B5 200F     01000          JR      NZ,M1     ;SI CNT<400
```

```
44B7 210000   01010          LD      HL,0      ;REINITIALISER COMPTEUR A ZERO
44BA 22D44D   01020          LD      (CNT),HL  ;
44BD 3AC64D   01030          LD      A,(P2)    ;DECREMENTER PARAMETRE P2
44C0 3D       01040          DEC     A         ;
44C1 2803     01050          JR      Z,M1      ;SI NUL NE PAS LE MODIFIER
44C3 32C64D   01060          LD      (P2),A    ;
44C6 3A4038   01070 M1       LD      A,(3840H) ;LECTURE CLAVIER
44C9 FE06     01080          CP      6         ;
44CB CA1044   01090          JP      Z,PRES    ;SI BREAK+CLEAR ARRETER LE JEU
44CE CD3F47   01100          CALL    ASM       ;AVANCE DES SOUS-MARINS
44D1 CD7345   01110          CALL    ABA       ;AVANCE DU BATEAU
44D4 3AC34D   01120          LD      A,(F1)    ;COMPTEUR DE DEPLACEMENT
44D7 3D       01130          DEC     A         ;
44D8 201C     01140          JR      NZ,Z2     ;SI NON NUL NE RIEN FAIRE
44DA CD1A46   01150          CALL    ABM       ;AVANCE DES BOMBES
44DD 3AC24D   01160          LD      A,(F0)    ;SECOND COMPTEUR
44E0 3D       01170          DEC     A         ;
44E1 200E     01180          JR      NZ,Z1     ;SI NON NUL NE RIEN FAIRE
44E3 CDEB48   01190          CALL    TSM       ;TIR DES SOUS-MARINS
44E6 CD3A49   01200          CALL    AMN       ;AVANCE DES MINES
44E9 CDD645   01210          CALL    TBA       ;TIR DU BATEAU
44EC CDF846   01220          CALL    SSM       ;SORTIE DES SOUS-MARINS
44EF 3E03     01230          LD      A,3       ;REINITIALISER
44F1 32C24D   01240 Z1       LD      (F0),A    ;COMPTEUR F0
44F4 3E02     01250          LD      A,2       ;
44F6 32C34D   01260 Z2       LD      (F1),A    ;ET COMPTEUR F1
44F9 210145   01270          LD      HL,R0     ;ADRESSE DE RETOUR
44FC E5       01290          PUSH    HL        ;EMPILER
44FD 2AD04D   01290          LD      HL,(W0)   ;ADR. DU S.P.
4500 E9       01300          JP      (HL)      ;A EXECUTER
4501 218044   01310 R0       LD      HL,Z0     ;ADR. DE RETOUR
4504 E5       01320          PUSH    HL        ;EMPILER
4505 2AD24D   01330          LD      HL,(W1)   ;ADR. DU S.P.
4508 E9       01340          JP      (HL)      ;A EXECUTER
              01350 ;
              01360 ;************ S O U S   P R O G R A M M E S *************
              01370 ;
              01380 ;************** D E S S I N   I N I T I A L **************
              01390 ;
4509 CDC749   01400 DES      CALL    CLS       ;EFFACEMENT ECRAN
450C 21803C   01410          LD      HL,DB+128 ;ADR. DEBUT DE LA MER
450F 0640     01420          LD      B,64      ;LARGEUR DE LA MER
4511 3E5F     01430          LD      A,5FH     ;CODE POUR DESSINER LA MER
4513 77       01440 A0       LD      (HL),A    ;DESSINER LA MER
4514 23       01450          INC     HL        ;
4515 10FC     01460          DJNZ    A0        ;
4517 11983C   01470          LD      DE,DB+152 ;POSITION BATEAU
451A ED53BF4D 01480          LD      (PBA),DE  ;DANS VARIABLE PBA
451E CDAF48   01490          CALL    DBA       ;DESSIN DU BATEAU
4521 21234D   01500          LD      HL,MSC    ;MESSAGE POUR LES SCORES
4524 11013C   01510          LD      DE,DB+1   ;EN PREMIERE LIGNE
4527 CD614A   01520          CALL    MES       ;AFFICHAGE
452A 2ACE4D   01530          LD      HL,(MC)   ;MEILLEUR SCORE
452D CD114A   01540          CALL    TRF       ;TRANSFORMER EN ASCII
4530 23       01550          INC     HL        ;
4531 11233C   01560          LD      DE,DB+35  ;
4534 CD614A   01570          CALL    MES       ;AFFICHER CE SCORE
4537 C9       01580          RET               ;
              01590 ;
              01600 ;************ I N I T I A L I S A T I O N S *************
              01610 ;
4538 3E01     01620 INIT     LD      A,1       ;
```

```
453A 32C24D   01630          LD      (F0),A   ;COMPTEUR DE DEPLACEMENT
453D 32C34D   01640          LD      (F1),A   ;IDEM
4540 210000   01650          LD      HL,0     ;
4543 22CC4D   01660          LD      (SC),HL  ;SCORE NUL
4546 22D44D   01670          LD      (CNT),HL ;COMPTEUR DE BOUCLE
4549 3E0C     01680          LD      A,12     ;
454B 32C14D   01690          LD      (PCA),A  ;CODE D'UNE BOMBE - 80H
454E 3EB4     01700          LD      A,180    ;
4550 32C44D   01710          LD      (P0),A   ;PARAMETRE SORTIE SOUS-MARINS
4553 3E04     01720          LD      A,4      ;
4555 32C54D   01730          LD      (P1),A   ;VALEUR EN POINTS MAX. D'UN S.M.
4558 3E14     01740          LD      A,20     ;
455A 32C64D   01750          LD      (P2),A   ;PARAMETRE TIR DES SOUS-MARINS
455D AF       01760          XOR     A        ;A=0
455E 32704D   01770          LD      (FPA),A  ;LA PARTIE COMMENCE
4561 32C74D   01780          LD      (FLG),A  ;LE BONUS N'EST PAS ATTEINT
4564 21714D   01790          LD      HL,PSM   ;DEBUT DES TABLEAUX DE VARIABLES
4567 064E     01800          LD      B,78     ;
4569 77       01810 K7       LD      (HL),A   ;MISE A ZERO DES TABLEAUX
456A 23       01820          INC     HL       ;POUR INDIQUER QUE TOUS LES
456B 10FC     01830          DJNZ    K7       ;MOBILES SONT INEXISTANTS
456D 3E03     01840          LD      A,3      ;
456F 32C94D   01850          LD      (NBA),A  ;NOMBRE INITIAL DE BATEAUX
4572 C9       01860          RET              ;
              01870 ;
              01880 ;********** A V A N C E   D U   B A T E A U **************
              01890 ;
4573 3A704D   01900 ABA      LD      A,(FPA)  ;
4576 B7       01910          OR      A        ;
4577 C0       01920          RET     NZ       ;SI FIN DE PARTIE RETOUR
4578 2ABF4D   01930          LD      HL,(PBA) ;POSITION BATEAU
457B 3ACA4D   01940          LD      A,(JEA)  ;EST-CE LE MODE DEMONSTRATION ?
457E B7       01950          OR      A        ;
457F 2829     01960          JR      Z,R5     ;SI MODE NORMAL
4581 3AFF38   01970          LD      A,(38FFH);LECTURE CLAVIER
4584 B7       01980          OR      A        ;
4585 C21044   01990          JP      NZ,PRES  ;SI FRAPPE D'UNE TOUCHE
4588 3E28     02000          LD      A,40     ;
458A CDDB49   02010          CALL    RND      ;POUR DEPLACEMENT ALEATOIRE
458D FE0A     02020          CP      10       ;
458F 2004     02030          JR      NZ,T4    ;SI A<>10
4591 3E20     02040          LD      A,20H    ;DEPLACEMENT A GAUCHE
4593 180D     02050          JR      T5       ;
4595 FE0F     02060 T4       CP      15       ;
4597 2004     02070          JR      NZ,T6    ;SI A<>15
4599 3E40     02080          LD      A,40H    ;DEPLACEMENT A DROITE
459B 1805     02090          JR      T5       ;
459D FE14     02100 T6       CP      20       ;
459F 2004     02110          JR      NZ,T7    ;SI A<>20 FAIRE LE MEME DEPLACEMENT
              02120                           ;QUE LE PRECEDENT
45A1 AF       02130          XOR     A        ;PAS DE DEPLACEMENT
45A2 32CB4D   02140 T5       LD      (DIR),A  ;SAUVEGARDER ANCIENNE COMMANDE
45A5 3ACB4D   02150 T7       LD      A,(DIR)  ;COMMANDE
45A8 1803     02160          JR      R6       ;EXECUTER CETTE COMMANDE
45AA 3A4038   02170 R5       LD      A,(3840H);LECTURE CLAVIER
45AD CB6F     02180 R6       BIT     5,A      ;
45AF 201C     02190          JR      NZ,GAU   ;SI FLECHE GAUCHE
45B1 CB77     02200          BIT     6,A      ;
45B3 C8       02210          RET     Z        ;SI PAS DE FLECHE DROITE
45B4 7D       02220 DRT      LD      A,L      ;DEPLACEMENT A DROITE
45B5 FEAF     02230          CP      175      ;
45B7 D0       02240          RET     NC       ;SI BATEAU BLOQUE RETOUR
```

```
45B8 23       02250          INC     HL        ;POSITION+1
45B9 E5       02260          PUSH    HL        ;
45BA 110F00   02270          LD      DE,15     ;
45BD 19       02280          ADD     HL,DE     ;ADR. DE FIN DU BATEAU
45BE 7E       02290          LD      A,(HL)    ;LECTURE CARACTERE
45BF D1       02300          POP     DE        ;ADR. DEBUT BATEAU
45C0 FE5F     02310 A9       CP      5FH       ;EST-CE LA MER ?
45C2 C27C4A   02320          JP      NZ,BOUM   ;NON EXPLOSION
45C5 ED53BF4D 02330          LD      (PBA),DE  ;NOUVELLE POSITION
45C9 CDAF48   02340          CALL    DBA       ;DESSIN BATEAU
45CC C9       02350          RET               ;
45CD 7D       02360 GAU      LD      A,L       ;DEPLACEMENT A GAUCHE
45CE FE82     02370          CP      130       ;
45D0 D8       02380          RET     C         ;SI BATEAU BLOQUE RETOUR
45D1 2B       02390          DEC     HL        ;POSITION-1
45D2 7E       02400          LD      A,(HL)    ;EST-CE LA MER ?
45D3 EB       02410          EX      DE,HL     ;POSITION DANS DE
45D4 18EA     02420          JR      A9        ;CONTINUER
              02430 ;
              02440 ;************ T I R   D U   B A T E A U *****************
              02450 ;
45D6 3A704D   02460 TBA      LD      A,(FPA)   ;EST-CE LA FIN DE LA PARTIE ?
45D9 B7       02470          OR      A         ;
45DA C0       02480          RET     NZ        ;SI OUI RETOUR
45DB 21B54D   02490          LD      HL,PBM    ;DEBUT DU TABLEAU DES BOMBES
45DE 0605     02500          LD      B,NBM     ;NOMBRE DE BOMBES
45E0 23       02510 L1       INC     HL        ;
45E1 7E       02520          LD      A,(HL)    ;OCTET DE POIDS FORT DE LA POSITION
45E2 B7       02530          OR      A         ;
45E3 2804     02540          JR      Z,J2      ;SI NUL PAS DE BOMBE
45E5 23       02550          INC     HL        ;
45E6 10F8     02560          DJNZ    L1        ;PASSER A L'ELEMENT SUIVANT
45E8 C9       02570          RET               ;
45E9 E5       02580 J2       PUSH    HL        ;SAUVEGARDER ADR. ELEMENT
45EA 2ABF4D   02590          LD      HL,(PBA)  ;POSITION BATEAU
45ED 23       02600          INC     HL        ;+1
45EE 3ACA4D   02610          LD      A,(JEA)   ;INDICATEUR JEU ALEATOIRE
45F1 B7       02620          OR      A         ;
45F2 2807     02630          JR      Z,R3      ;SI NUL MODE ORDINAIRE
45F4 3EFF     02640          LD      A,255     ;
45F6 CDDB49   02650          CALL    RND       ;SINON TIR ALEATOIRE
45F9 1803     02660          JR      R4        ;
45FB 3A0438   02670 R3       LD      A,(3804H) ;LECTURE CLAVIER
45FE CB4F     02680 R4       BIT     1,A       ;
4600 2008     02690          JR      NZ,J3     ;SI APPUI SUR <Q>
4602 CB7F     02700          BIT     7,A       ;
4604 2812     02710          JR      Z,M9      ;SI PAS APPUI SUR <W>
4606 110D00   02720          LD      DE,13     ;
4609 19       02730          ADD     HL,DE     ;ADR. DU BORD DROIT DU BATEAU
460A CBFC     02740 J3       SET     7,H       ;BIT 7 A 1 POUR INDIQUER LE TIR
460C EB       02750          EX      DE,HL     ;POSITION DANS DE
460D E1       02760          POP     HL        ;ADR. ELEMENT
460E 72       02770          LD      (HL),D    ;RANGER OCTET POIDS FORT POSITION
460F 2B       02780          DEC     HL        ;
4610 73       02790          LD      (HL),E    ;RANGER OCTET POIDS FAIBLE
4611 21464B   02800          LD      HL,SN5    ;ADR. DU S.P. DE SON POUR LE TIR
4614 22D04D   02810          LD      (W0),HL   ;REMPLACER LA DEMI-BOUCLE D'ATTENTE
4617 C9       02820          RET               ;
4618 E1       02830 M9       POP     HL        ;RESTAURER ADR.
4619 C9       02840          RET               ;
              02850 ;
              02860 ;********* A V A N C E   D E S   B O M B E S ************
```

```
                02870 ;
461A 3AC14D     02880 ABM     LD      A,(PCA)  ;CODE DU CARACTERE DES BOMBES
461D 07         02890         RLCA             ;DEPLACER LA BOMBE D'UN SEGMENT
461E 07         02900         RLCA             ;
461F CB7F       02910         BIT     7,A      ;
4621 2018       02920         JR      NZ,I2    ;SI BIT 7 A 1 SAUTER DE LIGNE
4623 32C14D     02930         LD      (PCA),A  ;RANGER CODE DU CARACTERE
4626 CBFF       02940         SET     7,A      ;+80H POUR OBTENIR LE VRAI CODE
4628 4F         02950         LD      C,A      ;CODE DANS C
4629 21B54D     02960         LD      HL,PBM   ;DEBUT TABLEAU DES BOMBES
462C 0605       02970         LD      B,NBM    ;NOMBRE DE BOMBES
462E 5E         02980 I1      LD      E,(HL)   ;OCTET DE POIDS FAIBLE POSITION
462F 23         02990         INC     HL       ;
4630 56         03000         LD      D,(HL)   ;OCTET DE POIDS FORT
4631 23         03010         INC     HL       ;
4632 7A         03020         LD      A,D      ;
4633 B7         03030         OR      A        ;
4634 2802       03040         JR      Z,I0     ;SI D=0 LA BOMBE N'EXISTE PAS
4636 79         03050         LD      A,C      ;CODE DU CARACTERE
4637 12         03060         LD      (DE),A   ;DESSINER LA BOMBE
4638 10F4       03070 I0      DJNZ    I1       ;PASSER A LA BOMBE SUIVANTE
463A C9         03080         RET              ;
463B 3E03       03090 I2      LD      A,3      ;SAUT DE LIGNE
463D 32C14D     03100         LD      (PCA),A  ;REINITIALISER CODE
4640 CBFF       03110         SET     7,A      ;+80H POUR VRAI CODE
4642 4F         03120         LD      C,A      ;CODE DANS C
4643 21B54D     03130         LD      HL,PBM   ;ADR. DEBUT TABLEAU DES BOMBES
4646 0605       03140         LD      B,NBM    ;NOMBRE DE BOMBES
4648 5E         03150 I3      LD      E,(HL)   ;OCTET DE POIDS FAIBLE POSITION
4649 23         03160         INC     HL       ;
464A 56         03170         LD      D,(HL)   ;OCTET DE POIDS FORT
464B 7A         03180         LD      A,D      ;
464C B7         03190         OR      A        ;
464D 281C       03200         JR      Z,I4     ;SI D=0 PAS DE BOMBE
464F CB7A       03210         BIT     7,D      ;
4651 CBBA       03220         RES     7,D      ;VRAI POSITION
4653 2004       03230         JR      NZ,M8    ;SI BIT 7 A 1 LA BOMBE VIENT
                03240                          ;D'ETRE TIREE, DONC NE RIEN EFFACER
4655 1A         03250         LD      A,(DE)   ;LECTURE DU CARACTERE
4656 E6CF       03260         AND     0CFH     ;EFFACER LA BOMBE
4658 12         03270         LD      (DE),A   ;NOUVEAU CARACTERE SANS LA BOMBE
4659 E5         03280 M8      PUSH    HL       ;SAUVEGARDER ADR. ELEMENT
465A 214000     03290         LD      HL,64    ;
465D 19         03300         ADD     HL,DE    ;PASSER A LA LIGNE SUIVANTE
465E EB         03310         EX      DE,HL    ;POSITION DANS DE
465F E1         03320         POP     HL       ;RESTAURER ADR. ELEMENT
4660 7A         03330         LD      A,D      ;
4661 FE40       03340         CP      40H      ;A-T-ON DEPASSE L'ECRAN ?
4663 380A       03350         JR      C,I5     ;SI NON
4665 1600       03360 J0      LD      D,0      ;ANNULER LA BOMBE
4667 72         03370 I6      LD      (HL),D   ;RANGER OCTET DE POIDS FORT
4668 2B         03380         DEC     HL       ;
4669 73         03390         LD      (HL),E   ;RANGER OCTET DE POIDS FAIBLE
466A 23         03400         INC     HL       ;
466B 23         03410 I4      INC     HL       ;
466C 10DA       03420         DJNZ    I3       ;PASSER A LA BOMBE SUIVANTE
466E C9         03430         RET              ;
466F 1A         03440 I5      LD      A,(DE)   ;LECTURE DE LA CASE OU LA BOMBE
                03450                          ;VA SE POSER
4670 FE80       03460         CP      80H      ;ESPACE ?
4672 280F       03470         JR      Z,I8     ;SI OUI
4674 FE80       03480         CP      0B0H     ;EST-CE UNE AUTRE BOMBE ?
```

```
4676 280B     03490          JR      Z,I8      ;SI OUI
4678 FE2A     03500          CP      '*'       ;EST-CE UNE MINE ?
467A 200B     03510          JR      NZ,I7     ;SI NON C'EST AUTRE CHOSE
467C E5       03520          PUSH    HL        ;SAUVEGARDER ADR. ELEMENT
467D C5       03530          PUSH    BC        ;SAUVEGARDER NOMBRE DE BOMBES
467E CDD747   03540          CALL    ETO       ;RENDRE LA MINE INVISIBLE
4681 C1       03550          POP     BC        ;
4682 E1       03560          POP     HL        ;
4683 79       03570 I8       LD      A,C       ;CODE DU CARACTERE
4684 12       03580          LD      (DE),A    ;DESSINER LA BOMBE
4685 18E0     03590          JR      I6        ;SUITE
4687 FE3A     03600 I7       CP      3AH       ;EST-CE UN CHIFFRE ?
4689 38DA     03610          JR      C,J0      ;SI OUI
468B E5       03620          PUSH    HL        ;C'EST UN SOUS-MARIN
468C C5       03630          PUSH    BC        ;
468D EB       03640          EX      DE,HL     ;
468E CD034A   03650          CALL    LIGNE     ;RECHERCHE NUMERO DE LIGNE
4691 D604     03660          SUB     4         ;
4693 47       03670          LD      B,A       ;
4694 87       03680          ADD     A,A       ;MULTIPLIER PAR 4
4695 87       03690          ADD     A,A       ;
4696 1600     03700          LD      D,0       ;
4698 5F       03710          LD      E,A       ;
4699 21714D   03720          LD      HL,PSM    ;ADR. DEBUT TABLE DES SOUS-MARINS
469C 19       03730          ADD     HL,DE     ;HL POINTE VERS LE SOUS-MARIN
              03740                            ;TOUCHE
469D CB40     03750          BIT     0,B       ;
469F 2805     03760          JR      Z,I9      ;SI SOUS-MARIN GAUCHE
46A1 CD5648   03770          CALL    ESD       ;EXPLOSION DU SOUS-MARIN DROIT
46A4 1803     03780          JR      J1        ;
46A6 CDF247   03790 I9       CALL    ESG       ;EXPLOSION DU SOUS-MARIN GAUCHE
46A9 C1       03800 J1       POP     BC        ;RESTAURER NOMBRE DE BOMBES
46AA E1       03810          POP     HL        ;ADR. ELEMENT
46AB 18B8     03820          JR      J0        ;
              03830 ;
              03840 ;******** SON POUR L'EXPLOSION D'UN SOUS-MARIN **********
              03850 ;
46AD C5       03860 SN0      PUSH    BC        ;
46AE 0EA0     03870          LD      C,160     ;COMPTEUR DE DUREE
46B0 3E01     03880          LD      A,1       ;COMMANDE SUR LE PORT
46B2 D3FF     03890 O0       OUT     (255),A   ;
46B4 41       03900          LD      B,C       ;
46B5 10FE     03910 O1       DJNZ    O1        ;ATTENTE VARIABLE
46B7 EE03     03920          XOR     3         ;PASSER DE 1 A 2 OU DE 2 A 1
46B9 08       03930          EX      AF,AF'    ;SAUVER COMMANDE
46BA 79       03940          LD      A,C       ;
46BB D605     03950          SUB     5         ;DECREMENTER COMPTEUR DE 5
46BD 2804     03960          JR      Z,O8      ;SI NUL RETOUR
46BF 4F       03970          LD      C,A       ;
46C0 08       03980          EX      AF,AF'    ;RESTAURER COMMANDE
46C1 18EF     03990          JR      O0        ;CONTINUER
46C3 C1       04000 O8       POP     BC        ;
46C4 C9       04010          RET               ;
              04020 ;
              04030 ;**** COMPTE DES POINTS GAGNES AVEC GESTION DU BONUS ****
              04040 ; EN ENTREE: HL CONTIENT LE NOUVEAU SCORE
              04050 ;
46C5 3AC74D   04060 CMT      LD      A,(FLG)   ;
46C8 B7       04070          OR      A         ;
46C9 201F     04080          JR      NZ,K5     ;SI BONUS DEJA ATTEINT
46CB 11C800   04090          LD      DE,200    ;
46CE DF       04100          RST     18H       ;COMPARER LE SCORE A 20000
```

```
46CF 3819     04110          JR      C,K5      ;SI INFERIEUR
46D1 3AC94D   04120          LD      A,(NBA)   ;NOMBRE DE BATEAUX
46D4 3C       04130          INC     A         ;+1
46D5 32C94D   04140          LD      (NBA),A   ;
46D8 C630     04150          ADD     A,30H     ;CONVERTIR EN ASCII
46DA 323E3C   04160          LD      (DB+62),A;AFFICHER CE NOMBRE
46DD 32C74D   04170          LD      (FLG),A   ;MODIFIER FLG
46E0 1E12     04180          LD      E,18      ;COMPTEUR DE BOUCLE POUR SONNERIE
46E2 163C     04190          LD      D,60      ;PARAMETRE POUR SN8
46E4 CD7C4B   04200 R9       CALL    SN8       ;SON BONUS
46E7 1D       04210          DEC     E         ;
46E8 20FA     04220          JR      NZ,R9     ;BOUCLER
46EA 22CC4D   04230 K5       LD      (SC),HL   ;MODIFIER SCORE
46ED CD114A   04240          CALL    TRF       ;TRANSFORMER EN ASCII
46F0 23       04250          INC     HL        ;
46F1 11093C   04260          LD      DE,DB+9   ;ADR. D'IMPLANTATION
46F4 CD614A   04270          CALL    MES       ;AFFICHAGE
46F7 C9       04280          RET               ;
              04290 ;
              04300 ;********** SORTIE ALEATOIRE DES SOUS-MARINS ***********
              04310 ;
46F8 21714D   04320 SSM      LD      HL,PSM    ;ADR. DEBUT TABLEAU DES SOUS-MARINS
46FB 060C     04330          LD      B,12      ;NOMBRE DE SOUS-MARINS
46FD 7E       04340 B4       LD      A,(HL)    ;COMPTEUR DE DEPLACEMENT
46FE B7       04350          OR      A         ;
46FF 2037     04360          JR      NZ,B0     ;SI NON NUL LE S.M. EST DEJA SORTI
4701 3AC44D   04370          LD      A,(P0)    ;PARAMETRE
4704 CDDB49   04380          CALL    RND       ;SORTIE ALEATOIRE
4707 FE01     04390          CP      1         ;
4709 202D     04400          JR      NZ,B0     ;SI A<> 1 LE S.M. NE SORT PAS
470B 3AC54D   04410          LD      A,(P1)    ;PARAMETRE POUR LE NOMBRE DE POINTS
470E CDDB49   04420          CALL    RND       ;
4711 3601     04430          LD      (HL),1    ;INITIALISER COMPTEUR
4713 23       04440          INC     HL        ;
4714 77       04450          LD      (HL),A    ;RANGER P1-VALEUR DU S.M.
4715 23       04460          INC     HL        ;
4716 3E0C     04470          LD      A,12      ;NBR. DE S.M.
4718 90       04480          SUB     B         ;NUMERO DE L'ELEMENT DANS LA TABLE
4719 E5       04490          PUSH    HL        ;SAUVER ADR. ELEMENT
471A 6F       04500          LD      L,A       ;
471B 2600     04510          LD      H,0       ;NUMERO DE L'ELEMENT DANS HL
471D 0E06     04520          LD      C,6       ;
471F 29       04530 B1       ADD     HL,HL     ;MULTIPLICATION PAR 64 POUR
4720 0D       04540          DEC     C         ;OBTENIR LA POSITION INITIALE
4721 20FC     04550          JR      NZ,B1     ;DU S.M.
4723 11403D   04560          LD      DE,DB+320 ;
4726 19       04570          ADD     HL,DE     ;POSITION DU S.M.
4727 CB40     04580          BIT     0,B       ;EST-CE UN S.M. GAUCHE
4729 2806     04590          JR      Z,B2      ;SI OUI
472B 114800   04600          LD      DE,72     ;
472E B7       04610          OR      A         ;
472F ED52     04620          SBC     HL,DE     ;SINON RETRANCHER 72 A LA POSITION
4731 EB       04630 B2       EX      DE,HL     ;POSITION DANS DE
4732 E1       04640          POP     HL        ;ADR. DE L'ELEMENT
4733 73       04650          LD      (HL),E    ;RANGER OCTET POIDS FAIBLE POSITION
4734 23       04660          INC     HL        ;
4735 72       04670          LD      (HL),D    ;OCTET DE POIDS FORT
4736 1803     04680          JR      B3        ;SUITE
4738 23       04690 B0       INC     HL        ;
4739 23       04700          INC     HL        ;
473A 23       04710          INC     HL        ;
473B 23       04720 B3       INC     HL        ;
```

```
473C 10BF      04730          DJNZ    B4        ;PASSER A L'ELEMENT SUIVANT
473E C9        04740          RET               ;
               04750 ;
               04760 ;***** A V A N C E   D E S   S O U S - M A R I N S ******
               04770 ;
473F 21714D    04780 ASM      LD      HL,PSM    ;ADR. DEBUT TABLEAU DES S.M.
4742 060C      04790          LD      B,12      ;NOMBRE DE S.M.
4744 7E        04800 B5       LD      A,(HL)    ;COMPTEUR DE DEPLACEMENT
4745 B7        04810          OR      A         ;
4746 2846      04820          JR      Z,B6      ;SI NUL PAS DE S.M.
4748 35        04830          DEC     (HL)      ;DECREMENTER COMPTEUR
4749 2043      04840          JR      NZ,B6     ;SI NON NUL NE PAS DEPLACER LE S.M.
474B E5        04850          PUSH    HL        ;SAUVER ADR. ELEMENT
474C 23        04860          INC     HL        ;
474D 7E        04870          LD      A,(HL)    ;P1-VALEUR DU S.M.
474E CB7F      04880          BIT     7,A       ;
4750 2043      04890          JR      NZ,C2     ;SI BIT 7 A 1 LE S.M. A EXPLOSE
4752 2B        04900          DEC     HL        ;
4753 77        04910          LD      (HL),A    ;REINITIALISER COMPTEUR
4754 23        04920          INC     HL        ;
4755 23        04930          INC     HL        ;
4756 5E        04940          LD      E,(HL)    ;OCTET DE POIDS FAIBLE POSITION
4757 23        04950          INC     HL        ;
4758 56        04960          LD      D,(HL)    ;OCTET DE POIDS FORT
4759 CB40      04970          BIT     0,B       ;
475B 2047      04980          JR      NZ,B7     ;SI S.M. DROIT
475D 1B        04990          DEC     DE        ;DEPLACER VERS LA GAUCHE
475E CB73      05000          BIT     6,E       ;POSITION VISIBLE ?
4760 280D      05010          JR      Z,B8      ;SI OUI
4762 210700    05020          LD      HL,7      ;
4765 19        05030          ADD     HL,DE     ;ADR. DE LA QUEUE DU S.M.
4766 CB75      05040          BIT     6,L       ;POSITION VISIBLE ?
4768 2819      05050          JR      Z,C0      ;SI OUI
476A E1        05060 C7       POP     HL        ;SI NON ANNULER S.M.
476B 3600      05070          LD      (HL),0    ;
476D 181F      05080          JR      B6        ;PASSER AU SUIVANT
476F 1A        05090 B8       LD      A,(DE)    ;LECTURE DU CARACTERE OU ON SE POSE
4770 FE80      05100          CP      80H       ;EST-CE UN ESPACE ?
4772 280F      05110          JR      Z,C0      ;SI OUI
4774 FE2A      05120          CP      '*'       ;EST-CE UNE MINE ?
4776 2005      05130          JR      NZ,C1     ;SI NON C'EST UNE BOMBE
4778 CDD747    05140          CALL    ETO       ;RENDRE LA MINE INVISIBLE
477B 1806      05150          JR      C0        ;
477D E1        05160 C1       POP     HL        ;RESTAURER ADR. ELEMENT
477E CDEF47    05170          CALL    ESGB      ;FAIRE EXPLOSER LE S.M.
4781 180B      05180          JR      B6        ;PASSER AU SUIVANT
4783 E1        05190 C0       POP     HL        ;ADR. ELEMENT
4784 E5        05200          PUSH    HL        ;
4785 23        05210          INC     HL        ;
4786 23        05220          INC     HL        ;
4787 73        05230          LD      (HL),E    ;RANGER POSITION
4788 23        05240          INC     HL        ;
4789 72        05250          LD      (HL),D    ;
478A CDDC48    05260          CALL    DSG       ;DESSINER LE S.M. GAUCHE
478D E1        05270 C4       POP     HL        ;ADR. ELEMENT
478E 23        05280 B6       INC     HL        ;
478F 23        05290          INC     HL        ;
4790 23        05300          INC     HL        ;
4791 23        05310          INC     HL        ;
4792 10B0      05320          DJNZ    B5        ;PASSER AU S.M. SUIVANT
4794 C9        05330          RET               ;
4795 23        05340 C2       INC     HL        ;TRAITEMENT APRES EXPLOSION
```

```
4796 5E        05350          LD      E,(HL)   ;OCTET DE POIDS FAIBLE POSITION
4797 23        05360          INC     HL       ;
4798 56        05370          LD      D,(HL)   ;OCTET DE POIDS FORT
4799 0E03      05380          LD      C,3      ;LARGEUR DU NOMBRE A EFFACER
479B 3E80      05390          LD      A,80H    ;ESPACE
479D 12        05400 C3       LD      (DE),A   ;EFFACER LE NOMBRE QUI APPARAIT
479E 13        05410          INC     DE       ;APRES L'EXPLOSION
479F 0D        05420          DEC     C        ;
47A0 20FB      05430          JR      NZ,C3    ;
47A2 18E9      05440          JR      C4       ;PASSER AU S.M. SUIVANT
47A4 13        05450 B7       INC     DE       ;DEPLACEMENT VERS LA DROITE
47A5 210700    05460          LD      HL,7     ;
47A8 19        05470          ADD     HL,DE    ;ADR. DE LA QUEUE DU S.M.
47A9 CB75      05480          BIT     6,L      ;POSITION VISIBLE ?
47AB 2006      05490          JR      NZ,C5    ;OUI
47AD CB73      05500          BIT     6,E      ;POSITION DE TETE VISIBLE ?
47AF 201A      05510          JR      NZ,C6    ;OUI
47B1 18B7      05520          JR      C7       ;NON ANNULER S.M.
47B3 7E        05530 C5       LD      A,(HL)   ;LECTURE DU CARACTERE OU ON SE POSE
47B4 FE80      05540          CP      80H      ;EST-CE UN ESPACE ?
47B6 2813      05550          JR      Z,C6     ;SI OUI
47B8 FE2A      05560          CP      '*'      ;EST-CE UNE MINE ?
47BA 2003      05570          JR      NZ,C8    ;SI NON C'EST UNE BOMBE
47BC EB        05580          EX      DE,HL    ;POSITION MINE DANS DE
47BD E5        05590          PUSH    HL       ;SAUVER POSITION S.M.
47BE CDD747    05600          CALL    ETO      ;RENDRE LA MINE INVISIBLE
47C1 D1        05610          POP     DE       ;POSITION S.M.
47C2 1807      05620          JR      C6       ;
47C4 EB        05630 C8       EX      DE,HL    ;POSITION BOMBE DANS DE
47C5 E1        05640          POP     HL       ;ADR. ELEMENT
47C6 CD5348    05650          CALL    ESDB     ;EXPLOSION DU S.M. A DROITE
47C9 18C3      05660          JR      B6       ;
47CB E1        05670 C6       POP     HL       ;ADR. ELEMENT
47CC E5        05680          PUSH    HL       ;
47CD 23        05690          INC     HL       ;
47CE 23        05700          INC     HL       ;
47CF 73        05710          LD      (HL),E   ;OCTET DE POIDS FAIBLE POSITION
47D0 23        05720          INC     HL       ;
47D1 72        05730          LD      (HL),D   ;OCTET DE POIDS FORT
47D2 CDCD48    05740          CALL    DSD      ;DESSIN DU S.M. DROIT
47D5 18B6      05750          JR      C4       ;PASSER AU S.M. SUIVANT
               05760 ;
               05770 ;********** S.P. RENDANT UNE MINE INVISIBLE **************
               05780 ; EN ENTREE: DE=POSITION DE LA MINE
               05790 ;
47D7 21A14D    05800 ETO      LD      HL,PMN   ;ADR. DEBUT DU TABLEAU DES MINES
47DA 0E05      05810          LD      C,NMN    ;NOMBRE DE MINES
47DC 7E        05820 G2       LD      A,(HL)   ;OCTET DE POIDS FAIBLE POSITION
47DD BB        05830          CP      E        ;EST-CE E ?
47DE 2009      05840          JR      NZ,G0    ;SI NON
47E0 23        05850          INC     HL       ;
47E1 7E        05860          LD      A,(HL)   ;OCTET DE POIDS FORT
47E2 CBBF      05870          RES     7,A      ;VRAI ADRESSE
47E4 BA        05880          CP      D        ;EST-CE D ?
47E5 2003      05890          JR      NZ,G1    ;SI NON
47E7 72        05900          LD      (HL),D   ;RANGER ADR. AVEC BIT 7 A ZERO
47E8 C9        05910          RET              ;
47E9 23        05920 G0       INC     HL       ;
47EA 23        05930 G1       INC     HL       ;
47EB 0D        05940          DEC     C        ;
47EC 20EE      05950          JR      NZ,G2    ;PASSER A LA MINE SUIVANTE
47EE C9        05960          RET              ;SI PAS TROUVEE
```

```
                05970 ;
                05980 ;*********** EXPLOSION DU SOUS-MARIN GAUCHE *************
                05990 ; EN ENTREE: HL CONTIENT L'ADR. DE LA VARIABLE DU S.M.
                06000 ;            DE CONTIENT L'ADR. DE LA BOMBE (POUR ESGB)
                06010 ;
47EF CD9148     06020 ESGB    CALL    EBM       ;ANNULER LA BOMBE
47F2 360A       06030 ESG     LD      (HL),10   ;COMPTEUR A 10, POUR AFFICHER LES
                06040                           ;POINTS GAGNES PENDANT 10 TOURS
47F4 E5         06050         PUSH    HL        ;SAUVER HL
47F5 C5         06060         PUSH    BC        ;ET BC
47F6 23         06070         INC     HL        ;
47F7 4E         06080         LD      C,(HL)    ;P1-VALEUR
47F8 CBFE       06090         SET     7,(HL)    ;BIT 7 A 1 POUR L'EXPLOSION
47FA 23         06100         INC     HL        ;
47FB 5E         06110         LD      E,(HL)    ;POIDS FAIBLE POSITION
47FC E5         06120         PUSH    HL        ;SAUVER ADR. ELEMENT
47FD 23         06130         INC     HL        ;
47FE 56         06140         LD      D,(HL)    ;POIDS FORT
47FF 2E07       06150         LD      L,7       ;LARGEUR DU S.M.
4801 CB73       06160 E1      BIT     6,E       ;
4803 2004       06170         JR      NZ,E0     ;SI CARACTERE INVISIBLE
4805 1A         06180         LD      A,(DE)    ;UN DES CARACTERES DU S.M.
4806 EE3F       06190         XOR     3FH       ;INVERSION VIDEO
4808 12         06200         LD      (DE),A    ;DESSINER CARACTERE INVERSE
4809 13         06210 E0      INC     DE        ;PASSER AU SUIVANT
480A 2D         06220         DEC     L         ;
480B 20F4       06230         JR      NZ,E1     ;BOUCLER
480D CDAD46     06240         CALL    SN0       ;SON EXPLOSION
4810 3E80       06250         LD      A,80H     ;ESPACE
4812 2E07       06260         LD      L,7       ;LARGEUR SU S.M.
4814 1B         06270 E3      DEC     DE        ;
4815 CB73       06280         BIT     6,E       ;
4817 2001       06290         JR      NZ,E2     ;SI CARACTERE CACHE
4819 12         06300         LD      (DE),A    ;EFFACER S.M.
481A 2D         06310 E2      DEC     L         ;
481B 20F7       06320         JR      NZ,E3     ;
481D 7B         06330         LD      A,E       ;
481E E67F       06340         AND     7FH       ;
4820 FE3E       06350         CP      62        ;
4822 3006       06360         JR      NC,T9     ;SI S.M. SUR LE BORD DROIT
4824 13         06370         INC     DE        ;SE PLACER AU MILIEU SU S.M.
4825 13         06380 E4      INC     DE        ;
4826 CB73       06390         BIT     6,E       ;SI CARACTERE INVISIBLE BOUCLER
4828 20FB       06400         JR      NZ,E4     ;JUSQU'A OBTENIR CARACTERE VISIBLE
482A 13         06410 T9      INC     DE        ;
482B 13         06420         INC     DE        ;PASSER AU DERNIER CHIFFRE
482C CB73       06430 E6      BIT     6,E       ;
482E 2803       06440         JR      Z,E5      ;SI VISIBLE
4830 1B         06450         DEC     DE        ;SINON DECALER LA NOMBRE JUSQU'A
4831 19F9       06460         JR      E6        ;CE QU'IL SOIT VISIBLE
4833 3E30       06470 E5      LD      A,'0'     ;
4835 12         06480         LD      (DE),A    ;AFFICHAGE DU TROISIEME CHIFFRE
4836 1B         06490         DEC     DE        ;
4837 12         06500         LD      (DE),A    ;SECOND CHIFFRE
4838 1B         06510         DEC     DE        ;
4839 E1         06520         POP     HL        ;ADR. DE L'ELEMENT
483A 73         06530         LD      (HL),E    ;kANGER POSITION
483B 23         06540         INC     HL        ;
483C 72         06550         LD      (HL),D    ;
483D 3AC54D     06560         LD      A,(P1)    ;PARAMETRE POUR LE NOMBRE DE POINTS
4840 91         06570         SUB     C         ;CALCUL DU NOMBRE DE POINTS GAGNE
4841 C631       06580         ADD     A,31H     ;CONVERTIR EN ASCII + 1
```

```
4843 12      06590          LD      (DE),A   ;AFFICHER
4844 D630    06600          SUB     30H      ;VRAI NOMBRE
4846 2ACC4D  06610          LD      HL,(SC)  ;SCORE
4849 5F      06620          LD      E,A      ;
484A 1600    06630          LD      D,0      ;
484C 19      06640          ADD     HL,DE    ;AUGMENTER LE SCORE
484D CDC546  06650          CALL    CMT      ;AFFICHER LE SCORE
4850 C1      06660          POP     BC       ;RESTAURER BC
4851 E1      06670          POP     HL       ;ET HL
4852 C9      06680          RET              ;
             06690 ;
             06700 ;************ EXPLOSION D'UN SOUS-MARIN DROIT ************
             06710 ; EN ENTREE: HL CONTIENT L'ADR. DE LA VARIABLE DU S.M.
             06720 ;            DE CONTIENT L'ADR. DE LA BOMBE (POUR ESDB)
             06730 ;
4853 CD9148  06740 ESDB     CALL    EBM      ;ANNULER LA BOMBE
4856 360A    06750 ESD      LD      (HL),10  ;COMPTEUR A 10
4858 E5      06760          PUSH    HL       ;SAUVER HL
4859 C5      06770          PUSH    BC       ;ET BC
485A 23      06780          INC     HL       ;
485B 4E      06790          LD      C,(HL)   ;P1-VALEUR
485C CBFE    06800          SET     7,(HL)   ;BIT 7 A 1 POUR L'EXPLOSION
485E 23      06810          INC     HL       ;
485F 5E      06820          LD      E,(HL)   ;POSITION S.M.
4860 E5      06830          PUSH    HL       ;SAUVER ADR. ELEMENT
4861 23      06840          INC     HL       ;
4862 56      06850          LD      D,(HL)   ;
4863 13      06860          INC     DE       ;SAUTER L'ESPACE EN BOUT DU S.M.
4864 2E07    06870          LD      L,7      ;LARGEUR DU S.M.
4866 CB73    06880 K1       BIT     6,E      ;
4868 2804    06890          JR      Z,K0     ;SI CARACTERE INVISIBLE
486A 1A      06900          LD      A,(DE)   ;
486B EE3F    06910          XOR     3FH      ;
486D 12      06920          LD      (DE),A   ;INVERSION VIDEO
486E 13      06930 K0       INC     DE       ;PASSER AU CARACTERE SUIVANT
486F 2D      06940          DEC     L        ;
4870 20F4    06950          JR      NZ,K1    ;
4872 CDAD46  06960          CALL    SN0      ;SON EXPLOSION
4875 3E30    06970          LD      A,30H    ;ESPACE
4877 2E07    06980          LD      L,7      ;LARGEUR DU S.M.
4879 1B      06990 K3       DEC     DE       ;
487A CB73    07000          BIT     6,E      ;
487C 2801    07010          JR      Z,K2     ;SI CARACTERE INVISIBLE
487E 12      07020          LD      (DE),A   ;EFFACER S.M.
487F 2D      07030 K2       DEC     L        ;
4880 20F7    07040          JR      NZ,K3    ;
4882 13      07050          INC     DE       ;SE PLACER AU MILIEU DU S.M.
4883 13      07060 K4       INC     DE       ;
4884 CB73    07070          BIT     6,E      ;
4886 28FB    07080          JR      Z,K4     ;SI CARACTERE INVISIBLE BOUCLER
4888 13      07090          INC     DE       ;
4889 13      07100          INC     DE       ;SE PLACER SUR LE DERNIER CHIFFRE
488A CB73    07110 K6       BIT     6,E      ;
488C 20A5    07120          JR      NZ,E5    ;SI VISIBLE
488E 1B      07130          DEC     DE       ;SINON DECALER LE NOMBRE
488F 18F9    07140          JR      K6       ;SUITE
             07150 ;
             07160 ;***** A N N U L A T I O N   D ' U N E   B O M B E ******
             07170 ; EN ENTREE: DE CONTIENT L'ADR. DE LA BOMBE
             07180 ;
4891 E5      07190 EBM      PUSH    HL       ;SAUVER HL
4892 C5      07200          PUSH    BC       ;ET BC
```

```
4893 21B54D   07210          LD      HL,PBM    ;ADR. DEBUT DU TABLEAU DES BOMBES
4896 0605     07220          LD      B,NBM     ;NOMBRE DE BOMBES
4898 7E       07230 J7       LD      A,(HL)    ;POIDS FAIBLE POSITION
4899 BB       07240          CP      E         ;EST-CE E ?
489A 200C     07250          JR      NZ,J4     ;SI NON PASSER A LA BOMBE SUIVANTE
489C 23       07260          INC     HL        ;
489D 7E       07270          LD      A,(HL)    ;POIDS FORT
489E BA       07280          CP      D         ;EST-CE D ?
489F 2008     07290          JR      NZ,J5     ;SI NON
48A1 3600     07300          LD      (HL),0    ;ANNULER BOMBE
48A3 3E80     07310          LD      A,80H     ;ESPACE
48A5 12       07320          LD      (DE),A    ;EFFACER BOMBE
48A6 1804     07330          JR      J6        ;RETOUR
48A8 23       07340 J4       INC     HL        ;
48A9 23       07350 J5       INC     HL        ;
48AA 10EC     07360          DJNZ    J7        ;PASSER A LA BOMBE SUIVANTE
48AC C1       07370 J6       POP     BC        ;RESTAURER BC
48AD E1       07380          POP     HL        ;ET HL
48AE C9       07390          RET               ;
              07400 ;
              07410 ;********** D E S S I N   D U   B A T E A U **********
              07420 ;
48AF 21984C   07430 DBA      LD      HL,BAT    ;ADR. CHAINE GRAPHIQUE
48B2 D5       07440 A4       PUSH    DE        ;ADR. DEBUT DE LIGNE
48B3 7E       07450 A3       LD      A,(HL)    ;CARACTERE DANS A
48B4 23       07460          INC     HL        ;
48B5 B7       07470          OR      A         ;
48B6 2813     07480          JR      Z,A1      ;SI NUL RETOUR
48B8 FE01     07490          CP      1         ;SI 1 SAUT DE LIGNE
48BA 2804     07500          JR      Z,A2      ;
48BC 12       07510          LD      (DE),A    ;AFFICHER CARACTERE
48BD 13       07520          INC     DE        ;POSITION SUIVANTE
48BE 18F3     07530          JR      A3        ;BOUCLER
48C0 E3       07540 A2       EX      (SP),HL   ;SAUTER A LA LIGNE PRECEDENTE
48C1 114000   07550          LD      DE,64     ;
48C4 B7       07560          OR      A         ;
48C5 ED52     07570          SBC     HL,DE     ;ADR. LIGNE PRECEDENTE
48C7 EB       07580          EX      DE,HL     ;POSITION DANS DE
48C8 E1       07590          POP     HL        ;ADR. CHAINE
48C9 18E7     07600          JR      A4        ;
48CB D1       07610 A1       POP     DE        ;RETOUR
48CC C9       07620          RET               ;
              07630 ;
              07640 ;** D E S S I N   D U   S O U S - M A R I N   D R O I T **
              07650 ;
48CD 21B64C   07660 DSD      LD      HL,SD     ;ADR. CHAINE GRAPHIQUE
48D0 7E       07670 A6       LD      A,(HL)    ;CARACTERE DANS A
48D1 23       07680          INC     HL        ;
48D2 B7       07690          OR      A         ;
48D3 C8       07700          RET     Z         ;SI NUL RETOUR
48D4 CB73     07710          BIT     6,E       ;
48D6 2801     07720          JR      Z,A5      ;SI BIT 6 A 0 NE PAS AFFICHER
48D8 12       07730          LD      (DE),A    ;AFFICHER LE CARACTERE
48D9 13       07740 A5       INC     DE        ;POSITION SUIVANTE
48DA 18F4     07750          JR      A6        ;CARACTERE SUIVANT
              07760 ;
              07770 ;* D E S S I N   D U   S O U S - M A R I N   G A U C H E *
              07780 ;
48DC 21BF4C   07790 DSG      LD      HL,SG     ;ADR. CHAINE GRAPHIQUE
48DF 7E       07800 A8       LD      A,(HL)    ;CARACTERE DANS A
48E0 23       07810          INC     HL        ;
48E1 B7       07820          OR      A         ;
```

```
48E2 C8       07830          RET     Z         ;SI NUL RETOUR
48E3 CB73     07840          BIT     6,E       ;
48E5 2001     07850          JR      NZ,A7     ;SI BIT 6 DE E A 1 NE PAS AFFICHER
48E7 12       07860          LD      (DE),A    ;AFFICHER CARACTERE
48E8 13       07870 A7       INC     DE        ;POSITION SUIVANTE
48E9 18F4     07880          JR      A8        ;PASSER AU CARACTERE SUIVANT
              07890 ;
              07900 ;******** T I R   D E S   S O U S - M A R I N S *********
              07910 ;
48EB 3A704D   07920 TSM      LD      A,(FPA)   ;EST-CE LA FIN DE LA PARTIE ?
48EE B7       07930          OR      A         ;
48EF C0       07940          RET     NZ        ;SI OUI
48F0 3AC64D   07950          LD      A,(P2)    ;PARAMETRE POUR LE TIR
48F3 CDDB49   07960          CALL    RND       ;TIR ALEATOIRE
48F6 FE01     07970          CP      1         ;
48F8 C0       07980          RET     NZ        ;SI A<>1 PAS DE TIR
48F9 3E0C     07990          LD      A,12      ;NOMBRE DE S.M.
48FB CDDB49   08000          CALL    RND       ;TIRER UN S.M. AU SORT
48FE 3D       08010          DEC     A         ;NUMERO DU S.M.
48FF 47       08020          LD      B,A       ;
4900 87       08030          ADD     A,A       ;MULTIPLIER A PAR 4
4901 87       08040          ADD     A,A       ;
4902 6F       08050          LD      L,A       ;
4903 2600     08060          LD      H,0       ;4*NUMERO DANS HL
4905 11714D   08070          LD      DE,PSM    ;ADR. DEBUT TABLE DES S.M.
4908 19       08080          ADD     HL,DE     ;ADR. VARIABLE DU S.M. CHOISI
4909 7E       08090          LD      A,(HL)    ;
490A B7       08100          OR      A         ;LE S.M. EXISTE-T-IL ?
490B C8       08110          RET     Z         ;SI NON RETOUR
490C 23       08120          INC     HL        ;
490D 23       08130          INC     HL        ;
490E 5E       08140          LD      E,(HL)    ;POIDS FAIBLE POSITION
490F 23       08150          INC     HL        ;
4910 56       08160          LD      D,(HL)    ;POIDS FORT
4911 13       08170          INC     DE        ;SE POSITIONNER AU MILIEU DU S.M.
4912 13       08180          INC     DE        ;
4913 13       08190          INC     DE        ;
4914 CB40     08200          BIT     0,B       ;
4916 2001     08210          JR      NZ,G3     ;SI S.M. DROIT
4918 13       08220          INC     DE        ;
4919 7B       08230 G3       LD      A,E       ;
491A E63F     08240          AND     3FH       ;
491C FE02     08250          CP      2         ;SI MINE TROP PRES DU BORD GAUCHE
491E D8       08260          RET     C         ;ON NE LA GENERE PAS
491F FE3E     08270          CP      62        ;
4921 D0       08280          RET     NC        ;IDEM POUR BORD DROIT
4922 21A14D   08290          LD      HL,PMN    ;ADR. DEBUT DU TABLEAU DES MINES
4925 0605     08300          LD      B,NMN     ;NOMBRE DE MINES
4927 23       08310 G4       INC     HL        ;
4928 7E       08320          LD      A,(HL)    ;POIDS FORT POSITION
4929 B7       08330          OR      A         ;
492A 200A     08340          JR      NZ,G5     ;SI NON NUL LA MINE EXISTE DEJA
492C 72       08350          LD      (HL),D    ;RANGER LA POSITION (BIT 7 A ZERO
492D 2B       08360          DEC     HL        ;LA MINE EST CACHEE PAR LE S.M.)
492E 73       08370          LD      (HL),E    ;POIDS FAIBLE
492F 21364B   08380          LD      HL,SN4    ;ADR. DU S.P. DE SON POUR LE TIR
4932 22D24D   08390          LD      (W1),HL   ;REMPLACER LA DEMI-BOUCLE D'ATTENTE
4935 C9       08400          RET               ;
4936 23       08410 G5       INC     HL        ;
4937 10EE     08420          DJNZ    G4        ;PASSER A L'ELEMENT SUIVANT
4939 C9       08430          RET               ;RETOUR SI TABLEAU DES MINES PLEIN
```

```
                08440 ;
                08450 ;********** A V A N C E   D E S   M I N E S **************
                08460 ;
493A 21A14D     08470 AMN     LD      HL,PMN    ;ADR. DEBUT TABLEAU DES MINES
493D 0605       08480         LD      B,NMN     ;NOMBRE DE MINES
493F 5E         08490 D4      LD      E,(HL)    ;POIDS FAIBLE POSITION
4940 23         08500         INC     HL        ;
4941 56         08510         LD      D,(HL)    ;POIDS FORT
4942 7A         08520         LD      A,D       ;
4943 B7         08530         OR      A         ;
4944 2866       08540         JR      Z,D5      ;SI D=0 PAS DE MINE
4946 FE01       08550         CP      1         ;
4948 284B       08560         JR      Z,D6      ;SI D=1 LA MINE VIENT D'EXPLOSER
494A CB7A       08570         BIT     7,D       ;
494C 2805       08580         JR      Z,D1      ;SI BIT 7 A 0 LA MINE EST CACHEE
494E CBBA       08590         RES     7,D       ;VERITABLE ADR.
4950 3E80       08600         LD      A,80H     ;ESPACE
4952 12         08610         LD      (DE),A    ;EFFACER LA MINE
4953 E5         08620 D1      PUSH    HL        ;SAUVER ADR. ELEMENT
4954 EB         08630         EX      DE,HL     ;POSITION DANS HL
4955 114000     08640         LD      DE,64     ;
4958 B7         08650         OR      A         ;
4959 ED52       08660         SBC     HL,DE     ;PASSER A LA LIGNE PRECEDENTE
495B CD034A     08670         CALL    LIGNE     ;CALCUL DU NUMERO DE LIGNE
495E EB         08680         EX      DE,HL     ;POSITION DANS DE
495F E1         08690         POP     HL        ;ADR. ELEMENT
4960 FE02       08700         CP      2         ;EST-CE LA LIGNE 2 ?
4962 2036       08710         JR      NZ,D2     ;SI NON
4964 1B         08720         DEC     DE        ;EXPLOSION A LA SURFACE
4965 1B         08730         DEC     DE        ;ADR. D'IMPLANTATION DE L'EXPLOSION
4966 E5         08740         PUSH    HL        ;SAUVER ADR. ELEMENT
4967 D5         08750         PUSH    DE        ;SAUVER POSITION
4968 21C84C     08760         LD      HL,EXM    ;CHAINE POUR DESSINER L'EXPLOSION
496B 0E05       08770         LD      C,5       ;LARGEUR DE LA CHAINE
496D 7E         08780 D7      LD      A,(HL)    ;CARACTERE DANS A
496E 12         08790         LD      (DE),A    ;AFFICHAGE
496F 23         08800         INC     HL        ;
4970 13         08810         INC     DE        ;POSITION SUIVANTE
4971 0D         08820         DEC     C         ;
4972 20F9       08830         JR      NZ,D7     ;DESSINER TOUS LES CARACTERES
4974 D1         08840         POP     DE        ;RESTAURER POSITION
4975 2ABF4D     08850         LD      HL,(PBA)  ;POSITION BATEAU
4978 B7         08860         OR      A         ;
4979 ED52       08870         SBC     HL,DE     ;SOUSTRAIRE LA POSITION DE LA MINE
497B 7D         08880         LD      A,L       ;
497C 3807       08890         JR      C,D9      ;SI BATEAU A GAUCHE DE LA MINE
497E FE05       08900         CP      5         ;SINON COMPARER A LA LARGEUR DE
                08910                           ;L'EXPLOSION DE LA MINE
4980 DA7C4A     08920         JP      C,BOUM    ;SI INFERIEUR LE BATEAU EST TOUCHE
4983 1805       08930         JR      D0        ;SINON
4985 FEF1       08940 D9      CP      0F1H      ;COMPARER A LA LARGEUR DU BATEAU
4987 D27C4A     08950         JP      NC,BOUM   ;SI INFERIEUR
498A 21664B     08960 D0      LD      HL,SN7    ;ADR. DU S.P. DE SON
498D 22D24D     08970         LD      (W1),HL   ;REMPLACER LA DEMI-BOUCLE D'ATTENTE
4990 E1         08980         POP     HL        ;ADR. DE L'ELEMENT
4991 1601       08990         LD      D,1       ;D=1 POUR INDIQUER L'EXPLOSION
4993 1813       09000         JR      D3        ;SUITE
4995 CDB949     09010 D6      CALL    EEM       ;EFFACEMENT DE L'EXPLOSION
4998 180E       09020         JR      D3        ;
499A 1A         09030 D2      LD      A,(DE)    ;LECTURE DE LA CASE OU ON SE POSE
499B FE2A       09040         CP      '*'       ;EST-CE UNE MINE ?
499D 2811       09050         JR      Z,U0      ;SI OUI
```

```
499F FE80     09060          CP      80H       ;EST-CE UN ESPACE ?
49A1 2005     09070          JR      NZ,D3     ;SI NON
49A3 3E2A     09080          LD      A,'*'     ;DESSINER LA MINE
49A5 12       09090          LD      (DE),A    ;
49A6 CBFA     09100 U1       SET     7,D       ;MINE VISIBLE
49A8 72       09110 D3       LD      (HL),D    ;RANGER POSITION MINE
49A9 2B       09120          DEC     HL        ;
49AA 73       09130          LD      (HL),E    ;
49AB 23       09140          INC     HL        ;
49AC 23       09150 D5       INC     HL        ;
49AD 1090     09160          DJNZ    D4        ;PASSER A LA MINE SUIVANTE
49AF C9       09170          RET               ;
49B0 E5       09180 U0       PUSH    HL        ;SAUVER ADR. ELEMENT
49B1 C5       09190          PUSH    BC        ;SAUVER NOMBRE DE MINES
49B2 CDD747   09200          CALL    ETO       ;RENDRE LA MINE INVISIBLE
49B5 C1       09210          POP     BC        ;
49B6 E1       09220          POP     HL        ;
49B7 18ED     09230          JR      U1        ;
              09240 ;
              09250 ;******** EFFACEMENT DE L'EXPLOSION DE LA MINE **********
              09260 ; EN ENTREE: E CONTIENT LA POSITION DE L'EXPLOSION DANS
              09270 ;                LA LIGNE
              09280 ;
49B9 163C     09290 EEM      LD      D,3CH     ;POIDS FORT POSITION
49BB 0E05     09300          LD      C,5       ;LARGEUR DE L'EXPLOSION
49BD 3E5F     09310          LD      A,5FH     ;CODE DE LA SURFACE DE LA MER
49BF 12       09320 D8       LD      (DE),A    ;REDESSINER SURFACE
49C0 13       09330          INC     DE        ;POSITION SUIVANTE
49C1 0D       09340          DEC     C         ;
49C2 20FB     09350          JR      NZ,D8     ;
49C4 1600     09360          LD      D,0       ;ANNULER LA MINE
49C6 C9       09370          RET               ;
              09380 ;
              09390 ;****** E F F A C E M E N T   D E   L ' E C R A N *******
              09400 ;
49C7 21003C   09410 CLS      LD      HL,3C00H ;DEBUT DE LA MEMOIRE ECRAN
49CA 11013C   09420          LD      DE,3C01H ;DEBUT DE LA MEMOIRE ECRAN +1
49CD 01FF03   09430          LD      BC,1023  ;TAILLE DE LA MEMOIRE ECRAN -1
49D0 3680     09440          LD      (HL),80H ;EFFACEMENT DU PREMIER CARACTERE
49D2 EDB0     09450          LDIR             ;EFFACEMENT DE TOUT L'ECRAN
49D4 C9       09460          RET
              09470 ;
              09480 ;******* INITIALISATION DU GENERATEUR ALEATOIRE *********
              09490 ;
49D5 ED5F     09500 RANDOM   LD      A,R      ;NOMBRE ALEATOIRE
49D7 32D64D   09510          LD      (SEED),A;INITIALISATION DU SEED
49DA C9       09520          RET
              09530 ;
              09540 ;*********** GENERATEUR DE NOMBRE ALEATOIRE ************
              09550 ; EN ENTREE: A CONTIENT LA LIMITE MAXIMALE DU NOMBRE
              09560 ;                ALEATOIRE
              09570 ; EN SORTIE: A CONTIENT UN NOMBRE ALEATOIRE COMPRIS ENTRE
              09580 ;                1 ET LE NOMBRE DONNE EN ENTREE
              09590 ; TOUS LES REGISTRES SONT SAUVEGARDES SAUF AF
              09600 ;
49DB D5       09610 RND      PUSH    DE       ;SAUVEGARDE DES REGISTRES DE
49DC E5       09620          PUSH    HL       ;ET HL
49DD 1600     09630          LD      D,0      ;LE NOMBRE MAXIMAL EST PLACE
49DF 5F       09640          LD      E,A      ;DANS DE
49E0 ED5F     09650          LD      A,R      ;LECTURE D'UN NOMBRE ALEATOIRE
              09660                           ;CONTENU DANS LE REGISTRE REFRESH
              09670                           ;A EST COMPRIS ENTRE 0 ET 127
```

```
49E2 6F       09680         LD      L,A      ;CE NOMBRE EST SAUVEGARDE DANS L
49E3 3AD64D   09690         LD      A,(SEED);LECTURE DU SEED PRECEDENT
49E6 AD       09700         XOR     L        ;CALCUL DU SEED SUIVANT
49E7 17       09710         RLA              ;A VARIE ENTRE 0 ET 254
49E8 6F       09720         LD      L,A      ;SAUVEGARDE DANS L
49E9 ED5F     09730         LD      A,R      ;NOMBRE ALEATOIRE
49EB AD       09740         XOR     L        ;CALCUL DU SEED SUIVANT
49EC 32D64D   09750         LD      (SEED),A;NOUVEAU SEED
49EF CDF749   09760         CALL    MUL      ;MULTIPLIER LE NOMBRE ALEATOIRE
              09770                          ;AVEC LE NOMBRE MAXIMAL DONNE
49F2 7C       09780         LD      A,H      ;NOMBRE ALEATOIRE ENTRE 0 ET
              09790                          ;LE NOMBRE MAXIMAL MOINS 1
49F3 3C       09800         INC     A        ;NOMBRE ALEATOIRE ENTRE 1 ET
              09810                          ;LE NOMBRE MAXIMAL
49F4 E1       09820         POP     HL       ;RESTAURATION DES REGISTRES DE
49F5 D1       09830         POP     DE       ;ET HL
49F6 C9       09840         RET
              09850 ;
              09860 ;********** SOUS PROGRAMME DE MULTIPLICATION ************
              09870 ;
49F7 210000   09880 MUL     LD      HL,0     ;
49FA 29       09890 M0      ADD     HL,HL    ;
49FB CB27     09900         SLA     A        ;
49FD 3001     09910         JR      NC,M1    ;
49FF 19       09920         ADD     HL,DE    ;
4A00 20F8     09930 M1      JR      NZ,M0    ;
4A02 C9       09940         RET              ;
              09950 ;
              09960 ;* SOUS PROGRAMME DONNANT LE NUMERO DE LIGNE D'UNE CASE *
              09970 ;********* MEMOIRE DONT ON CONNAIT L'ADRESSE ************
              09980 ; EN ENTREE: HL CONTIENT L'ADRESSE DE LA CASE MEMOIRE
              09990 ; EN SORTIE: A  CONTIENT LE NUMERO DE LIGNE
              10000 ;               COMPRIS ENTRE 0 ET 15
              10010 ; TOUS LES REGISTRES SAUF AF SONT SAUVEGARDES
              10020 ;
4A03 D5       10030 LIGNE   PUSH    DE       ;SAUVEGARDER DE
4A04 E5       10040         PUSH    HL       ;ET HL
4A05 11003C   10050         LD      DE,3C00H;DEBUT DE LA MEMOIRE ECRAN
4A08 B7       10060         OR      A        ;MISE A ZERO DU CARRY
4A09 ED52     10070         SBC     HL,DE    ;HL CONTIENT L'ADRESSE RELATIVE DE
              10080                          ;LA CASE PAR RAPPORT AU DEBUT DE
              10090                          ;LA MEMOIRE ECRAN
4A0B 29       10100         ADD     HL,HL    ;MULTIPLIER CETTE ADRESSE PAR 2
4A0C 29       10110         ADD     HL,HL    ;PAR 4
4A0D 7C       10120         LD      A,H      ;A CONTIENT LE NUMERO DE LIGNE
              10130                          ;( 0 <= A <= 15 )
4A0E E1       10140         POP     HL       ;RESTAURER HL
4A0F D1       10150         POP     DE       ;ET DE
4A10 C9       10160         RET
              10170 ;
              10180 ;* TRANSFORMATION D'UN NOMBRE ENTIER SIGNE CODE SUR 16 *
              10190 ;* BITS EN UNE SUITE DE CARACTERES ASCII LE REPRESENTANT *
              10200 ;******** LE NOMBRE PEUT VARIER ENTRE 0 ET 65535 ********
              10210 ; EN ENTREE: HL CONTIENT LE NOMBRE A CONVERTIR
              10220 ; EN SORTIE: HL CONTIENT L'ADRESSE DU PREMIER ELEMENT DE
              10230 ;               LA CHAINE REPRESENTANT LE NOMBRE
              10240 ;               IL EST ALORS POSSIBLE D'UTILISER LE SOUS
              10250 ;               PROGRAMME MES POUR L'AFFICHAGE
              10260 ;
4A11 DD21D84D 10270 TRF     LD      IX,BUF+1   ;IX SERT A REMPLIR LE BUFFER
              10280                          ;QUI VA CONTENIR LA CHAINE
4A15 DD36FF20 10290         LD      (IX-1),20H ;PAS DE SIGNE PAR DEFAUT
```

```
4A19 111027    10300 TRD     LD      DE,10000   ;
4A1C CD3F4A    10310         CALL    CAR        ;GARDER LE NOMBRE DES DIZAINES
               10320                            ;DE MILLE
4A1F 11E803    10330         LD      DE,1000    ;
4A22 CD3F4A    10340         CALL    CAR        ;GARDER LE NOMBRE DES MILLES
4A25 116400    10350         LD      DE,100     ;
4A28 CD3F4A    10360         CALL    CAR        ;GARDER LE NOMBRE DES CENTAINES
4A2B 110A00    10370         LD      DE,10      ;
4A2E CD3F4A    10380         CALL    CAR        ;GARDER LE NOMBRE DES UNITES
4A31 7D        10390         LD      A,L        ;L CONTIENT LE NOMBRE DES UNITES
4A32 C630      10400         ADD     A,30H      ;TRANSFORMER EN ASCII
4A34 DD7700    10410         LD      (IX+0),A   ;MISE DANS LE BUFFER A LA SUITE
4A37 DD360100  10420         LD      (IX+1),0   ;METTRE UN ZERO A LA FIN
4A3B 21D84D    10430         LD      HL,BUF+1   ;HL POINTE VERS LE PREMIER
               10440                            ;ELEMENT
4A3E C9        10450         RET
               10460 ;
               10470 ;*** SOUS PROGRAMME DIVISANT HL PAR DE ET PLACANT LE ****
               10480 ;************ QUOTIENT+30H DANS LE BUFFER ***************
               10490 ;
4A3F B7        10500 CAR     OR      A          ;MISE A ZERO DU CARRY
4A40 06FF      10510         LD      B,255      ;B=-1
4A42 ED52      10520 G9      SBC     HL,DE      ;ON SOUSTRAIT HL PAR DE JUSQU'A
               10530                            ;CE QU'IL Y AIT DEBORDEMENT
4A44 04        10540         INC     B          ;INCREMENTER LE QUOTIENT
4A45 30FB      10550         JR      NC,G9      ;SI PAS DEBORDEMENT
4A47 19        10560         ADD     HL,DE      ;HL CONTIENT LE RESTE
4A48 78        10570         LD      A,B        ;A CONTIENT LE QUOTIENT
4A49 C630      10580         ADD     A,30H      ;TRANSFORMATION EN ASCII
4A4B DD7700    10590         LD      (IX+0),A;MISE DANS LE BUFFER
4A4E DD23      10600         INC     IX         ;INCREMENTER LE POINTEUR DU BUFFER
4A50 C9        10610         RET
               10620 ;
               10630 ;***** DETERMINATION DE LA PRESENCE DES MINUSCULES ******
               10640 ; FLAG VAUT 0 S'IL N'Y A PAS DE MINUSCULES
               10650 ;       ET 0AFH DANS LE CAS CONTRAIRE
               10660 ;
4A51 21003C    10670 PRMN    LD      HL,3C00H ;ADR. CASE 1 DE L'ECRAN
4A54 3661      10680         LD      (HL),61H ;LA MINUSCULE a Y EST PLACEE
4A56 7E        10690         LD      A,(HL)   ;LECTURE DE LA VALEUR MEMORISEE
4A57 FE61      10700         CP      61H      ;EST-CE BIEN a ?
4A59 2001      10710         JR      NZ,0E+1  ;SI NON METTRE FLAG A ZERO
               10720                          ;EN EXECUTANT XOR  A (CODE 0AFH)
4A5B 3EAF      10730 0E      LD      A,0AFH   ;METTRE LE FLAG A 0AFH
4A5D 32C84D    10740         LD      (FLAG),A ;POSITIONNEMENT DU FLAG
4A60 C9        10750         RET
               10760 ;
               10770 ;* AFFICHAGE D'UN MESSAGE SUR L'ECRAN EN TENANT COMPTE **
               10780 ;**** DE LA PRESENCE OU DE L'ABSENCE DES MINUSCULES *****
               10790 ; EN ENTREE: HL CONTIENT L'ADRESSE DU PREMIER ELEMENT
               10800 ;               DU MESSAGE A AFFICHER
               10810 ;               LE MESSAGE DOIT SE TERMINER PAR UN OCTET 0
               10820 ;            DE CONTIENT L'ADRESSE A PARTIR DE LAQUELLE
               10830 ;             IL DOIT ETRE POSITIONNE
               10840 ; LES REGISTRES HL, DE ET AF SONT MODIFIES
               10850 ;
4A61 7E        10860 MES     LD      A,(HL)   ;LECTURE DU CARACTERE A AFFICHER
4A62 B7        10870         OR      A        ;EST-CE UN ZERO ?
4A63 C8        10880         RET     Z        ;SI OUI LE MESSAGE EST TERMINE
4A64 CD6C4A    10890         CALL    MIN      ;CONVERSION EN MAJUSCULE SI LES
               10900                          ;MINUSCULES NE SONT PAS INSTALLEES
4A67 12        10910         LD      (DE),A   ;AFFICHAGE D'UN CARACTERE
```

```
4A68 13        10920          INC     DE       ;INCREMENTATION DES POINTEURS DE
4A69 23        10930          INC     HL       ;ET HL
4A6A 18F5      10940          JR      MES      ;PASSER AU CARACTERE SUIVANT
               10950 ;
               10960 ;* SOUS PROGRAMME TRANSFORMANT UNE MINUSCULE EN MAJUSCULE *
               10970 ;* SI LES MINUSCULES NE SONT PAS INSTALLEES SUR LE TRS 80 *
               10980 ; EN ENTREE: A CONTIENT LE CARACTERE A TESTER
               10990 ; EN SORTIE: A CONTIENT LE CARACTERE MODIFIE
               11000 ;
4A6C FE7B      11010 MIN      CP      7BH      ;LE CODE ASCII EST-IL SUPERIEUR
               11020                           ;A CELUI DE z ?
4A6E D0        11030          RET     NC       ;SI OUI RETOUR
4A6F FE61      11040          CP      61H      ;LE CODE EST-IL INFERIEUR A a ?
4A71 D8        11050          RET     C        ;SI OUI RETOUR
4A72 47        11060          LD      B,A      ;SAUVEGARDE DANS B
4A73 3AC84D    11070          LD      A,(FLAG);LECTURE DU FLAG
4A76 B7        11080          OR      A        ;EST-IL NUL ?
4A77 78        11090          LD      A,B      ;RESTAURER LE CARACTERE
4A78 C0        11100          RET     NZ       ;SI FLAG=0 RETOUR (MINUSCULES)
4A79 CBAF      11110          RES     5,A      ;CONVERSION EN MAJUSCULE
4A7B C9        11120          RET
               11130 ;
               11140 ;********** EXPLOSION DU BATEAU DU JOUEUR **************
               11150 ;
4A7C ED5BBF4D  11160 BOUM     LD      DE,(PBA) ;POSITION BATEAU
4A80 21CD4C    11170          LD      HL,BA1   ;ADR. PREMIERE CHAINE GRAPHIQUE
4A83 D5        11180          PUSH    DE       ;SAUVEGARDE ADR.
4A84 CDB248    11190          CALL    A4       ;DESSIN DU NAVIRE SOMBRANT (1)
4A87 D1        11200          POP     DE       ;ADR. BATEAU
4A88 CD204B    11210          CALL    SN1      ;SON EXPLOSION
4A8B 13        11220          INC     DE       ;
4A8C 13        11230          INC     DE       ;SECOND DESSIN MOINS LARGE
4A8D 21EE4C    11240          LD      HL,BA2   ;ADR. SECOND DESSIN
4A90 CDB248    11250          CALL    A4       ;DESSIN DU BATEAU SOMBRANT (2)
4A93 CD204B    11260          CALL    SN1      ;SON EXPLOSION
4A96 21A14D    11270          LD      HL,PMN   ;DEBUT DU TABLEAU DES MINES
4A99 0605      11280          LD      B,NMN    ;NOMBRE DE MINES
4A9B 5E        11290 L2       LD      E,(HL)   ;POIDS FAIBLE ADR. MINE
4A9C 23        11300          INC     HL       ;
4A9D 56        11310          LD      D,(HL)   ;POIDS FORT
4A9E 7A        11320          LD      A,D      ;
4A9F B7        11330          OR      A        ;SI D=0 PAS DE MINE
4AA0 280B      11340          JR      Z,L3     ;PASSER A L'ELEMENT SUIVANT
4AA2 3600      11350          LD      (HL),0   ;MINE DETRUITE
4AA4 CB7F      11360          BIT     7,A      ;MINE CACHEE ?
4AA6 2805      11370          JR      Z,L3     ;OUI
4AA8 CBBA      11380          RES     7,D      ;ADR. DE LA MINE DANS DE
4AAA 3E80      11390          LD      A,80H    ;ESPACE
4AAC 12        11400          LD      (DE),A   ;EFFACER LA MINE
4AAD 23        11410 L3       INC     HL       ;PASSER A L'ELEMENT SUIVANT
4AAE 10EB      11420          DJNZ    L2       ;TRAITER TOUTES LES MINES
4AB0 21B54D    11430          LD      HL,PBM   ;DEBUT DU TABLEAU DES BOMBES
4AB3 0605      11440          LD      B,NBM    ;NOMBRE DE BOMBES
4AB5 5E        11450 L4       LD      E,(HL)   ;POIDS FAIBLE ADR. BOMBE
4AB6 23        11460          INC     HL       ;
4AB7 56        11470          LD      D,(HL)   ;POIDS FORT
4AB8 CBBA      11480          RES     7,D      ;ADR. DE LA BOMBE DANS DE
4ABA 7A        11490          LD      A,D      ;
4ABB B7        11500          OR      A        ;
4ABC 2805      11510          JR      Z,L5     ;SI D=0 PAS DE BOMBE
4ABE 3E80      11520          LD      A,80H    ;ESPACE
4AC0 12        11530          LD      (DE),A   ;EFFACER BOMBE
```

```
4AC1 3600      11540          LD      (HL),0    ;DETRUIRE BOMBE
4AC3 23        11550 L5       INC     HL        ;PASSER A L'ELEMENT SUIVANT
4AC4 10EF      11560          DJNZ    L4        ;TRAITER TOUTES LES BOMBES
4AC6 21803C    11570          LD      HL,DB+128;ADR. DEBUT DE LA MER
4AC9 0640      11580          LD      B,64      ;LARGEUR DE LA MER
4ACB 3E5F      11590          LD      A,5FH     ;
4ACD 77        11600 L6       LD      (HL),A    ;DESSINER LA MER
4ACE 23        11610          INC     HL        ;
4ACF 10FC      11620          DJNZ    L6        ;
4AD1 CD204B    11630          CALL    SN1       ;SON EXPLOSION
4AD4 11983C    11640          LD      DE,DB+152;ADR. POSITION BATEAU
4AD7 ED53BF4D  11650          LD      (PBA),DE  ;DANS VARIABLE PBA
4ADB CDAF48    11660          CALL    DBA       ;DESSIN BATEAU
4ADE 3E01      11670          LD      A,1       ;
4AE0 32C24D    11680          LD      (F0),A    ;REINITIALISER COMPTEURS
4AE3 32C34D    11690          LD      (F1),A    ;
4AE6 3E0C      11700          LD      A,12      ;
4AE8 32C14D    11710          LD      (PCA),A   ;
4AEB 3AC94D    11720          LD      A,(NBA)   ;NOMBRE DE BATEAUX
4AEE 3D        11730          DEC     A         ;
4AEF 32C94D    11740          LD      (NBA),A   ;1 DE MOINS
4AF2 C630      11750          ADD     A,30H     ;CONVERTIR EN ASCII
4AF4 323E3C    11760          LD      (DB+62),A;AFFICHAGE
4AF7 FE30      11770          CP      30H       ;EST-CE 0 ?
4AF9 C28044    11780          JP      NZ,Z0     ;NON CONTINUER LE JEU
4AFC 2ACC4D    11790 L7       LD      HL,(SC)   ;SCORE
4AFF ED5BCE4D  11800          LD      DE,(MC)   ;MEILLEUR SCORE
4B03 DF        11810          RST     18H       ;COMPARAISON
4B04 3803      11820          JR      C,M0      ;SI PAS MEILLEUR
4B06 22CE4D    11830          LD      (MC),HL   ;NOUVEAU RECORD
4B09 11593C    11840 M0       LD      DE,DB+89  ;ADR. D'IMPLANTATION
4B0C 21624D    11850          LD      HL,TER    ;ADR. MESSAGE 'TERMINE'
4B0F CD614A    11860          CALL    MES       ;AFFICHAGE
4B12 3E01      11870          LD      A,1       ;
4B14 32704D    11880          LD      (FPA),A   ;INDICATEUR FIN DE PARTIE
4B17 210000    11890          LD      HL,0      ;
4B1A 22D44D    11900          LD      (CNT),HL  ;REINITIALISER COMPTEUR
4B1D C39044    11910          JP      Z0        ;ALLER A LA BOUCLE PRINCIPALE
               11920                            ;POUR UN CERTAIN TEMPS
               11930 ;
               11940 ;********** SON POUR EXPLOSION DU BATEAU **************
               11950 ;
4B20 D5        11960 SN1      PUSH    DE        ;
4B21 1650      11970          LD      D,80      ;COMPTEUR DE DUREE
4B23 3E01      11980          LD      A,1       ;COMMANDE DU PORT
4B25 D3FF      11990 O2       OUT     (255),A   ;
4B27 08        12000          EX      AF,AF'    ;SAUVER COMMANDE
4B28 4A        12010          LD      C,D       ;
4B29 0601      12020          LD      B,1       ;
4B2B CD6000    12030          CALL    60H       ;ATTENTE VARIABLE
4B2E 08        12040          EX      AF,AF'    ;COMMANDE
4B2F EE03      12050          XOR     3         ;PASSER DE 1 A 2 OU DE 2 A 1
4B31 15        12060          DEC     D         ;
4B32 20F1      12070          JR      NZ,O2     ;BOUCLER
4B34 D1        12080          POP     DE        ;
4B35 C9        12090          RET               ;
               12100 ;
               12110 ;************** SON POUR TIR DU SOUS-MARIN ***************
               12120 ;
4B36 0E48      12130 SN4      LD      C,72      ;COMPTEUR DE DUREE
4B38 3E01      12140          LD      A,1       ;COMMANDE DU PORT
4B3A D3FF      12150 O3       OUT     (255),A   ;
```

```
4B3C 0623      12160          LD      B,35      ;
4B3E 10FE      12170 Q4       DJNZ    Q4        ;ATTENTE FIXE
4B40 EE03      12180          XOR     3         ;
4B42 0D        12190          DEC     C         ;
4B43 20F5      12200          JR      NZ,Q3     ;
4B45 C9        12210          RET               ;
               12220 ;
               12230 ;*************** SON POUR TIR DU BATEAU ******************
               12240 ;
4B46 0E80      12250 SN5      LD      C,128     ;COMPTEUR DUREE
4B48 3E01      12260 Q7       LD      A,1       ;COMMANDE PORT
4B4A D3FF      12270          OUT     (255),A   ;
4B4C 41        12280          LD      B,C       ;
4B4D 10FE      12290 Q5       DJNZ    Q5        ;DUREE VARIABLE
4B4F 3E03      12300          LD      A,3       ;COMMANDE DU PORT
4B51 D3FF      12310          OUT     (255),A   ;
4B53 79        12320          LD      A,C       ;
4B54 2F        12330          CPL               ;CALCUL DE LA SECONDE ATTENTE
4B55 47        12340          LD      B,A       ;
4B56 10FE      12350 Q6       DJNZ    Q6        ;ATTENTE
4B58 79        12360          LD      A,C       ;
4B59 D608      12370          SUB     8         ;DECREMENTER DE 8 LE COMPTEUR
4B5B 4F        12380          LD      C,A       ;
4B5C 20EA      12390          JR      NZ,Q7     ;BOUCLER
4B5E C9        12400          RET               ;
               12410 ;
               12420 ;*************** DEMI-BOUCLE D'ATTENTE ********************
               12430 ;
4B5F 010005    12440 SN6      LD      BC,500H   ;
4B62 CD6000    12450          CALL    60H       ;ATTENTE
4B65 C9        12460          RET               ;
               12470 ;
               12480 ;************ SON POUR EXPLOSION DES MINES ***************
               12490 ;
4B66 0E82      12500 SN7      LD      C,130     ;COMPTEUR DE DUREE
4B68 3E01      12510          LD      A,1       ;COMMANDE PORT
4B6A D3FF      12520 R2       OUT     (255),A   ;
4B6C 41        12530          LD      B,C       ;
4B6D 10FE      12540 T3       DJNZ    T3        ;DUREE VARIABLE
4B6F EE03      12550          XOR     3         ;
4B71 08        12560          EX      AF,AF'    ;
4B72 79        12570          LD      A,C       ;
4B73 C605      12580          ADD     A,5       ;AJOUTER 5 AU COMPTEUR
4B75 FEC8      12590          CP      200       ;
4B77 C8        12600          RET     Z         ;S'IL VAUT 200 ARRETER
4B78 4F        12610          LD      C,A       ;
4B79 08        12620          EX      AF,AF'    ;
4B7A 18EE      12630          JR      R2        ;BOUCLER
               12640 ;
               12650 ;************ SON POUR BONUS ET PRESENTATION *************
               12660 ;
4B7C 4A        12670 SN8      LD      C,D       ;COMPTEUR DE DUREE
4B7D 3E01      12680          LD      A,1       ;COMMANDE PORT
4B7F D3FF      12690 R7       OUT     (255),A   ;
4B81 41        12700          LD      B,C       ;
4B82 10FE      12710 R8       DJNZ    R8        ;DUREE VARIABLE
4B84 EE03      12720          XOR     3         ;
4B86 0D        12730          DEC     C         ;
4B87 20F6      12740          JR      NZ,R7     ;BOUCLER
4B89 C9        12750          RET               ;
               12760 ;
               12770 ;* CHAINE GRAPHIQUE UTILISEE POUR DESSINER LE NOM DU JEU *
```

```
                12780 ;
                12790 ;** LETTRE S **
4B8A A0B8       12800 SONAR   DEFW    0B8A0H
4B8C BC9F       12810         DEFW    9FBCH
4B8E 8F8F       12820         DEFW    8F8FH
4B90 8FBF       12830         DEFW    0BF8FH
4B92 BC8F       12840         DEFW    8FBCH
4B94 BF91       12850         DEFW    91BFH
4B96 8080       12860         DEFW    8080H
4B98 8080       12870         DEFW    8080H
4B9A 8383       12880         DEFW    8383H
4B9C 828B       12890         DEFW    8B82H
4B9E AFBC       12900         DEFW    0BCAFH
4BA0 B080       12910         DEFW    80B0H
4BA2 8080       12920         DEFW    8080H
4BA4 8080       12930         DEFW    8080H
4BA6 8080       12940         DEFW    8080H
4BA8 8283       12950         DEFW    8382H
4BAA 8FBD       12960         DEFW    0BD8FH
4BAC B490       12970         DEFW    90B4H
4BAE B0B0       12980         DEFW    0B0B0H
4BB0 8080       12990         DEFW    8080H
4BB2 8080       13000         DEFW    8080H
4BB4 A2BF       13010         DEFW    0BFA2H
4BB6 BF8F       13020         DEFW    8FBFH
4BB8 BFBC       13030         DEFW    0BCBFH
4BBA BCBC       13040         DEFW    0BCBCH
4BBC BE8F       13050         DEFW    8FBEH
4BBE 8781       13060         DEFW    8187H
                13070 ;** LETTRE O **
4BC0 B0BC       13080         DEFW    0BCB0H
4BC2 BF8F       13090         DEFW    8FBFH
4BC4 838F       13100         DEFW    8F83H
4BC6 BFBC       13110         DEFW    0BCBFH
4BC8 B0BF       13120         DEFW    0BFB0H
4BCA BF80       13130         DEFW    80BFH
4BCC 8080       13140         DEFW    8080H
4BCE 8080       13150         DEFW    8080H
4BD0 BFBF       13160         DEFW    0BFBFH
4BD2 BFBF       13170         DEFW    0BFBFH
4BD4 8080       13180         DEFW    8080H
4BD6 8080       13190         DEFW    8080H
4BD8 80BF       13200         DEFW    0BF80H
4BDA BFBF       13210         DEFW    0BFBFH
4BDC BF80       13220         DEFW    80BFH
4BDE 8080       13230         DEFW    8080H
4BE0 8080       13240         DEFW    8080H
4BE2 BFBF       13250         DEFW    0BFBFH
4BE4 BFBF       13260         DEFW    0BFBFH
4BE6 B080       13270         DEFW    80B0H
4BE8 8080       13280         DEFW    8080H
4BEA B0BF       13290         DEFW    0BFB0H
4BEC BF83       13300         DEFW    83BFH
4BEE 8FBF       13310         DEFW    0BF8FH
4BF0 BFBC       13320         DEFW    0BCBFH
4BF2 BFBF       13330         DEFW    0BFBFH
4BF4 8F83       13340         DEFW    838FH
                13350 ;** LETTRE N **
4BF6 BFBF       13360         DEFW    0BFBFH
4BF8 9080       13370         DEFW    8090H
4BFA 8080       13380         DEFW    8080H
4BFC 80BF       13390         DEFW    0BF80H
```

```
4BFE BFBF     13400          DEFW    0BFBFH
4C00 BFBF     13410          DEFW    0BFBFH
4C02 B490     13420          DEFW    90B4H
4C04 8080     13430          DEFW    8080H
4C06 BFBF     13440          DEFW    0BFBFH
4C08 BFBF     13450          DEFW    0BFBFH
4C0A 8BBF     13460          DEFW    0BF8BH
4C0C BD90     13470          DEFW    90BDH
4C0E 80BF     13480          DEFW    0BF80H
4C10 BFBF     13490          DEFW    0BFBFH
4C12 BF80     13500          DEFW    80BFH
4C14 82AF     13510          DEFW    0AF82H
4C16 BFB4     13520          DEFW    0B4BFH
4C18 BFBF     13530          DEFW    0BFBFH
4C1A BFBF     13540          DEFW    0BFBFH
4C1C 8080     13550          DEFW    8080H
4C1E 888B     13560          DEFW    8B80H
4C20 BFBF     13570          DEFW    0BFBFH
4C22 BFBF     13580          DEFW    0BFBFH
4C24 BF80     13590          DEFW    80BFH
4C26 8080     13600          DEFW    8080H
4C28 8082     13610          DEFW    8280H
4C2A BFBF     13620          DEFW    0BFBFH
              13630 ;** LETTRE A **
4C2C B0BC     13640          DEFW    0BCB0H
4C2E BF8F     13650          DEFW    8FBFH
4C30 838F     13660          DEFW    8F83H
4C32 BFBC     13670          DEFW    0BCBFH
4C34 B0BF     13680          DEFW    0BFB0H
4C36 BF80     13690          DEFW    80BFH
4C38 8080     13700          DEFW    8080H
4C3A 8080     13710          DEFW    8080H
4C3C BFBF     13720          DEFW    0BFBFH
4C3E BFBF     13730          DEFW    0BFBFH
4C40 B0B0     13740          DEFW    0B0B0H
4C42 B0B0     13750          DEFW    0B0B0H
4C44 B0BF     13760          DEFW    0BFB0H
4C46 BFBF     13770          DEFW    0BFBFH
4C48 BF83     13780          DEFW    83BFH
4C4A 8383     13790          DEFW    8383H
4C4C 8383     13800          DEFW    8383H
4C4E BFBF     13810          DEFW    0BFBFH
4C50 BFBF     13820          DEFW    0BFBFH
4C52 8080     13830          DEFW    8080H
4C54 8080     13840          DEFW    8080H
4C56 80BF     13850          DEFW    0BF80H
4C58 BFBF     13860          DEFW    0BFBFH
4C5A BF80     13870          DEFW    80BFH
4C5C 8080     13880          DEFW    8080H
4C5E 8080     13890          DEFW    8080H
4C60 BFBF     13900          DEFW    0BFBFH
              13910 ;** LETTRE R **
4C62 B0BC     13920          DEFW    0BCB0H
4C64 BF8F     13930          DEFW    8FBFH
4C66 838F     13940          DEFW    8F83H
4C68 BFBC     13950          DEFW    0BCBFH
4C6A B0BF     13960          DEFW    0BFB0H
4C6C BF80     13970          DEFW    80BFH
4C6E 8080     13980          DEFW    8080H
4C70 8080     13990          DEFW    8080H
4C72 BFBF     14000          DEFW    0BFBFH
4C74 BFBF     14010          DEFW    0BFBFH
```

```
4C76 BFBC     14020          DEFW    0BCBFH
4C78 B0BC     14030          DEFW    0BCB0H
4C7A BFBF     14040          DEFW    0BFBFH
4C7C 8FBF     14050          DEFW    0BF8FH
4C7E BF83     14060          DEFW    83BFH
4C80 8BBF     14070          DEFW    0BF8BH
4C82 BF93     14080          DEFW    93BFH
4C84 8080     14090          DEFW    8080H
4C86 BFBF     14100          DEFW    0BFBFH
4C88 8080     14110          DEFW    8080H
4C8A 82AF     14120          DEFW    0AF82H
4C8C BFB4     14130          DEFW    0B4BFH
4C8E 80BF     14140          DEFW    0BF80H
4C90 BF80     14150          DEFW    80BFH
4C92 8080     14160          DEFW    8080H
4C94 808B     14170          DEFW    8B80H
4C96 BFBD     14180          DEFW    0BDBFH
              14190 ;
              14200 ;************ DESSIN DU BATEAU DU JOUEUR ****************
              14210 ;
4C98 5F83     14220 BAT      DEFW    835FH
4C9A AFBF     14230          DEFW    0BFAFH
4C9C BFBF     14240          DEFW    0BFBFH
4C9E BFBF     14250          DEFW    0BFBFH
4CA0 BFBF     14260          DEFW    0BFBFH
4CA2 BFBF     14270          DEFW    0BFBFH
4CA4 BF9F     14280          DEFW    9FBFH
4CA6 835F     14290          DEFW    5F83H
4CA8 01       14300          DEFB    1         ;SAUT DE LIGNE
4CA9 80       14310          DEFB    80H
4CAA 8080     14320          DEFW    8080H
4CAC 8080     14330          DEFW    8080H
4CAE B09C     14340          DEFW    9CB0H
4CB0 BEBD     14350          DEFW    0BDBEH
4CB2 ACB0     14360          DEFW    0B0ACH
4CB4 8000     14370          DEFW    80H
              14380 ;
              14390 ;******** DESSIN DU SOUS-MARIN ALLANT A DROITE **********
              14400 ;
4CB6 808C     14410 SD       DEFW    8C80H
4CB8 8F8F     14420          DEFW    8F8FH
4CBA 8C8C     14430          DEFW    8C8CH
4CBC 8C8C     14440          DEFW    8C8CH
4CBE 00       14450          DEFB    0
              14460 ;
              14470 ;******** DESSIN DU SOUS-MARIN ALLANT A GAUCHE **********
              14480 ;
4CBF 8C8C     14490 SG       DEFW    8C8CH
4CC1 8C8C     14500          DEFW    8C8CH
4CC3 8F8F     14510          DEFW    8F8FH
4CC5 8C80     14520          DEFW    808CH
4CC7 00       14530          DEFB    0
              14540 ;
              14550 ;***** DESSIN DE L'EXPLOSION DES MINES A LA SURFACE *****
              14560 ;
4CC8 8C89     14570 EXM      DEFW    898CH
4CCA B096     14580          DEFW    96B0H
4CCC 9C       14590          DEFB    9CH
              14600 ;
              14610 ;******* DESSINS DU BATEAU PENDANT SON EXPLOSION ********
              14620 ;
4CCD 5F5F     14630 BA1      DEFW    5F5FH
```

```
4CCF 828F      14640          DEFW    8F82H
4CD1 BFBC      14650          DEFW    0BCBFH
4CD3 B7BF      14660          DEFW    0BFB7H
4CD5 BFBB      14670          DEFW    0BBBFH
4CD7 BCBF      14680          DEFW    0BFBCH
4CD9 8F81      14690          DEFW    818FH
4CDB 5F5F      14700          DEFW    5F5FH
4CDD 01        14710          DEFB    1
4CDE 8080      14720          DEFW    8080H
4CE0 A0BC      14730          DEFW    0BCA0H
4CE2 9080      14740          DEFW    8090H
4CE4 80A0      14750          DEFW    0A080H
4CE6 9080      14760          DEFW    8090H
4CE8 80A0      14770          DEFW    0A080H
4CEA BC90      14780          DEFW    90BCH
4CEC 8000      14790          DEFW    80H
4CEE 5F5F      14800 BA2      DEFW    5F5FH
4CF0 8AB4      14810          DEFW    0B48AH
4CF2 A0B8      14820          DEFW    0B8A0H
4CF4 B490      14830          DEFW    90B4H
4CF6 B885      14840          DEFW    85B8H
4CF8 5F5F      14850          DEFW    5F5FH
4CFA 01        14860          DEFB    1
4CFB 8080      14870          DEFW    8080H
4CFD 8080      14880          DEFW    8080H
4CFF 8080      14890          DEFW    8080H
4D01 8080      14900          DEFW    8080H
4D03 8080      14910          DEFW    8080H
4D05 8080      14920          DEFW    8080H
4D07 00        14930          DEFB    0
               14940 ;
               14950 ;******************** MESSAGES DIVERS ********************
               14960 ;
4D08 54        14970 TAP      DEFM    'TAPEZ ENTER POUR COMMENCER'
     41 50 45 5A 20 45 4E 54
     45 52 20 50 4F 55 52 20
     43 4F 4D 4D 45 4E 43 45
     52
4D22 00        14980          DEFB    0
4D23 53        14990 MSG      DEFM    'SCORE = 000000   MEILLEUR SCORE = 000000
BATEAUX RESTANTS: 3'
     43 4F 52 45 20 3D 20 30
     30 30 30 30 30 20 20 20
     4D 45 49 4C 4C 45 55 52
     20 53 43 4F 52 45 20 3D
     20 30 30 30 30 30 30 20
     20 20 42 41 54 45 41 55
     58 20 52 45 53 54 41 4E
     54 53 3A 20 33
4D61 00        15000          DEFB    0
4D62 54        15010 TER      DEFM    'T E R M I N E'
     20 45 20 52 20 4D 20 49
     20 4E 20 45
4D6F 00        15020          DEFB    0
               15030 ;
               15040 ;*************** VARIABLES ET TABLEAUX ******************
               15050 ;
0001           15060 FPA      DEFS    1        ;INDICATEUR FIN DE PARTIE
0030           15070 PSM      DEFS    48       ;POSITION DES SOUS-MARINS
0014           15080 PMN      DEFS    20       ;POSITION DES MINES
000A           15090 PBM      DEFS    10       ;POSITION DES BOMBES
0002           15100 PBA      DEFS    2        ;POSITION DU BATEAU
0001           15110 PCA      DEFS    1        ;CARACTERE POUR LES BOMBES - 80H
```

```
0001            15120 F0      DEFS    1         ;COMPTEUR DE DEPLACEMENT
0001            15130 F1      DEFS    1         ;COMPTEUR DE DEPLACEMENT
0001            15140 P0      DEFS    1         ;PARAMETRE POUR LA SORTIE DES
                15150                           ;SOUS-MARINS
0001            15160 P1      DEFS    1         ;PARAMETRE DONNANT LA VALEUR
                15170                           ;MAXIMALE D'UN SOUS-MARIN
0001            15180 P2      DEFS    1         ;PARAMETRE POUR LE TIR DES MINES
0001            15190 FLG     DEFS    1         ;INDICATEUR POUR LE BONUS
                15200                           ;VAUT 0 SI BONUS NON ENCORE ATTEINT
0001            15210 FLAG    DEFS    1         ;PRESENCE DES MINUSCULES
0001            15220 NBA     DEFS    1         ;NOMBRE DE BATEAUX RESTANTS
0001            15230 JEA     DEFS    1         ;INDICATEUR MODE DEMONSTRATION
                15240                           ;VAUT 0 SI MODE NORMAL, 1 SINON
0001            15250 DIR     DEFS    1         ;DERNIERE COMMANDE POUR LE
                15260                           ;MODE DEMONSTRATION
0002            15270 SC      DEFS    2         ;SCORE
0002            15280 MC      DEFS    2         ;MEILLEUR SCORE
0002            15290 W0      DEFS    2         ;PREMIERE SOUS-BOUCLE D'ATTENTE
0002            15300 W1      DEFS    2         ;SECONDE SOUS-BOUCLE D'ATTENTE
0002            15310 CNT     DEFS    2         ;NOMBRE DE PASSAGES DANS LA BOUCLE
                15320                           ;PRINCIPALE
0001            15330 SEED    DEFS    1         ;VARIABLE POUR RND
0007            15340 BUF     DEFS    7         ;BUFFER CONTENANT LA CHAINE
4DDE            15350 ZZ      EQU     $
4400            15360         END     DEBUT
00000 Total errors
```

Annuaire des utilisateurs du TRS 80

Vous possédez un TRS 80 qui vous passionne. Ses applications ne sont limitées que par votre imagination. Vous aimeriez contacter d'autres utilisateurs pour partager cette passion, échanger vos programmes, vos expériences, vos idées. Les contacts sont toujours très enrichissants et sont nécessaires pour assurer un certain équilibre car l'évasion de l'esprit que procure la pratique des micro-ordinateurs a tendance à renforcer notre isolement. La pratique solitaire des micro-ordinateurs est à la longue frustrante ! Elle suscite un tel enthousiasme que des clubs ont été créés. Mais tout le monde n'adhère pas forcément à un club ou à une association.

D'autres utilisateurs solitaires partageant votre passion habitent peut-être dans votre ville ou dans votre rue !

Comment saisir ces opportunités ? Nous vous proposons la constitution d'un annuaire des utilisateurs du TRS 80. Pour vous faire connaître de tous les autres utilisateurs, vous désirez y figurer ? Il suffit de retourner le formulaire ci-joint (1) (ou une photocopie) complété. L'annuaire paraîtra dès qu'un nombre suffisant de réponses nous sera parvenu.

(1) Retour du formulaire aux Éditions Eyrolles, 61 boulevard Saint-Germain - 75005 Paris.

Comment vous joindre ?

NOM .....
PRÉNOM .....
ADRESSE .....
CODE POSTAL .....

Téléphone .....

Facultatif: **Quelle configuration possédez-vous ?** *(cocher la case correspondante).*

Modèle ? | I | III |

Basic "Level" ? | 1 | 2 | 3 |

Capacité mémoire ? | 4 K | 16 K | 32 K | 48 K |

Combien d'unités de disques ? | 0 | 1 | 2 | 3 |

Imprimante ? | OUI | NON |

table traçante ? | OUI | NON |

Utilisation ? | Jeux | Calcul | Commande d'interfaces | Gestion |

Possédez-vous des interfaces ou adaptations ?

sonores	graphique haute résolution	option Z 80 4 MHZ	Joysticks	liaison RS 232	Modem	autre

Construisez-vous des interfaces ? | OUI | NON |

Imprimerie de la Manutention – Mayenne
Dépôt légal : mai 1982
N° d'Éditeur : 3784

Vous connaissez sans doute ces jeux vidéo qui font fureur dans les cafés et qui mettent à l'épreuve votre dextérité manuelle. Il est parfaitement possible de les transposer sur le TRS 80 et même d'en inventer de nouveaux.

Vous trouverez dans cet ouvrage toutes les techniques permettant de programmer vous même ces jeux sur votre ordinateur. Vous apprendrez de façon pratique à déplacer des graphismes en basse ou haute résolution sur l'écran, à produire des sonorités spéciales et à organiser votre programme de façon à gérer toutes les actions simultanément.

Ce livre entièrement original est actuellement le seul sur le sujet. Il a été réalisé par l'auteur de divers jeux commercialisés tant en France qu'aux États-Unis.

EYROLLES

8602

EYROLLES

JEUX D'ACTION RAPIDE SUR TRS 80

P. PELLIER